

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **11272534 A**

(43) Date of publication of application: **08.10.99**

(51) Int. Cl.

G06F 12/00

G06F 15/16

(21) Application number: **10163109**

(22) Date of filing: **11.06.98**

(30) Priority: **20.01.98 JP 10 8154**

(71) Applicant: **FUJITSU LTD**

(72) Inventor: **OKUBO TAKAO
HARA HIROTAKE
MATSUZUKA TAKAHIDE**

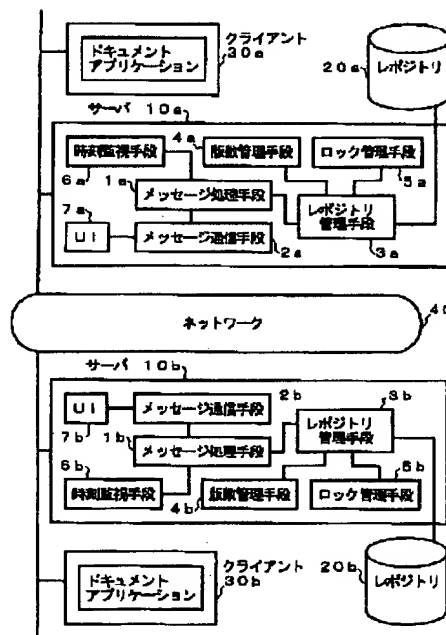
(54) **DOCUMENT DISTRIBUTION PROCESSING METHOD, SERVER MANAGEMENT METHOD FOR THE SAME AND RECORDING MEDIUM FOR SERVER PROGRAM**

(57) Abstract:

PROBLEM TO BE SOLVED: To enable reference and update at any base and to solve a problem of a multiple update occurrence in a document distribution processing in a system where an online document is shared by plural servers on a network.

SOLUTION: A server 10a for updating a document has a message communication means 2a transmit an update start notification to another server 10b. The server 10b receives this and has a lock management means 5b lock the document concerned. When document update is completed by the server 10a, the message communication means 2a transmits the updated document or its meta information to the server 10b. The server 10b releases the document lock at the time of receiving the updated document or the like. At that time, update start before a lock completion notification, distribution of a logical object only, and revision recognition by a server of high priority are performed.

COPYRIGHT: (C)1999,JPO



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-272534

(43) 公開日 平成11年(1999)10月8日

(51) Int.Cl.⁹

G 0 6 F 12/00
15/16

識別記号

5 3 3
3 7 0

F I

G 0 6 F 12/00
15/16

5 3 3 J
3 7 0 M

審査請求 未請求 請求項の数18 O L (全 33 頁)

(21) 出願番号 特願平10-163109

(22) 出願日 平成10年(1998)6月11日

(31) 優先権主張番号 特願平10-8154

(32) 優先日 平10(1998)1月20日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番
1号

(72) 発明者 大久保 隆夫

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(72) 発明者 原 裕貴

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(72) 発明者 松塚 貴英

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(74) 代理人 弁理士 小笠原 吉義 (外2名)

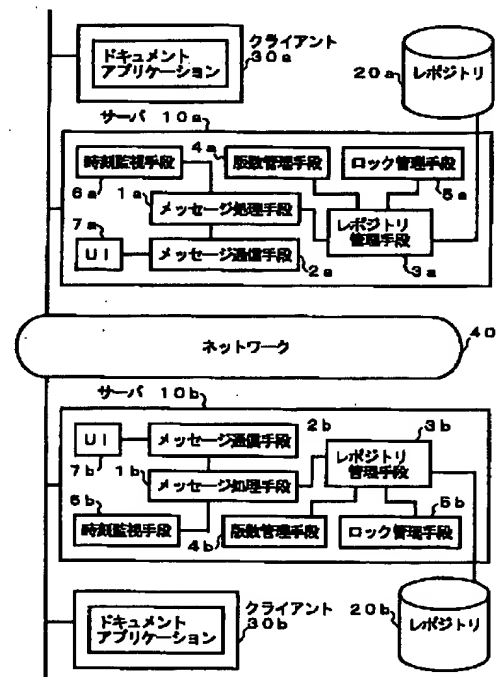
(54) 【発明の名称】 ドキュメント分散処理方法、ドキュメント分散処理システムのサーバ管理方法およびサーバプログラム
の記録媒体

(57) 【要約】

【課題】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有するシステムにおけるドキュメント分散処理方法に関し、どの拠点においても参照・更新を可能とし、かつ多重更新発生の問題を解決することを目的とする。

【解決手段】 文書を更新するサーバ10aでは、メッセージ通信手段2aにより他のサーバ10bに対して更新開始通知を送信する。サーバ10bはこれを受け取り、ロック管理手段5bにより該当する文書にロックをする。サーバ10aで文書更新を終了すると、メッセージ通信手段2aにより更新文書またはそのメタ情報がサーバ10bに送信される。サーバ10bでは、更新文書等を受け取った時点で文書ロックを解除する。このとき、ロック完了通知前の更新開始、論理オブジェクトのみの配布、優先度の高いサーバによる改版承認の処理を行う。

本発明の原理説明図



【特許請求の範囲】

【請求項1】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、同一のドキュメントの複製を各サーバに配布し、あるドキュメントの更新を開始するサーバが他のサーバに対して更新開始通知を送り、更新が終了したときに更新終了通知を送り、前記他のサーバは、前記更新開始通知を受け取ったときに該当するドキュメントに対する更新をロックし、更新終了通知を受け取ったときに前記ロックを解除することを特徴とするドキュメント分散処理方法。

【請求項2】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、同一のドキュメントの複製を各サーバに配布し、あるサーバがあるドキュメントを更新する場合に、そのドキュメントの更新開始と同時に他のサーバに対して更新開始通知を送り、更新開始通知に対する前記他のサーバからの承認の応答を受け取る前に、そのドキュメントの更新を進め、前記他のサーバからの更新の承認の有無は、更新が終了するまでの間に判定することを特徴とするドキュメント分散処理方法。

【請求項3】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、ドキュメントをその内容である実体と、その実体にリンクするメタ情報を持つ論理オブジェクトとに分離し、ドキュメントを作成または更新した場合に、他のサーバに対して論理オブジェクトの複製のみを送信し、その後、他のサーバからの請求または所定の契機で、前記作成または更新された実体の複製を送信することを特徴とするドキュメント分散処理方法。

【請求項4】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、各サーバに優先度を設定するとともに、各サーバごとに前記優先度に対応する独自のドキュメント版体系を設定し、ドキュメントを更新するサーバは、最も優先度の高いサーバに対し改版申請を送信し、改版が承認された場合にその版を正式版とすることを特徴とするドキュメント分散処理方法。

【請求項5】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、各サーバに優先度を設定するとともに、各サーバごとに前記優先度に対応する独自のドキュメント版体系を設定し、ドキュメントを更新するサーバは、前記優先度が最も高いサーバに対して改版申請を送信し、前記優先度が最も高いサーバとの通信が途絶している場合に

は、前記通信が途絶したサーバの次に高い優先度を持つサーバに対して改版申請を送信し、改版が承認された版を正式版とすることを特徴とするドキュメント分散処理方法。

【請求項6】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、システムへの新規サーバの登録依頼があった場合に、任意の依頼されたサーバは、依頼された時点における共有ドキュメントの複製を、新規サーバへ配送するとともに、その後に更新されたドキュメントのデータの新規サーバへの配送を、全サーバへの新規サーバの登録が完了するまで代行することを特徴とするドキュメント分散処理システムのサーバ管理方法。

【請求項7】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、あるドキュメントの管理情報を管理する管理サーバを他のサーバに変更する場合に、元の管理サーバは、依頼されたドキュメントの管理情報を新しい管理サーバへ配送するとともに、そのドキュメントの管理情報に対する他のサーバからの要求を、新しい管理サーバに対して転送し、新しい管理サーバは、前記ドキュメントの管理情報を取得した後、その管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知することを特徴とするドキュメント分散処理システムのサーバ管理方法。

【請求項8】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、あるドキュメントの管理情報を管理する管理サーバをシステムから削除する場合に、削除される管理サーバは、前記ドキュメントの管理情報の管理を引き継ぐサーバに対して、そのドキュメントの管理情報を配送するとともに、そのドキュメントの管理情報に対する他のサーバからの要求を、管理を引き継ぐ管理サーバに対して転送し、管理を引き継ぐ管理サーバからの引き継ぎの確認後に他のサーバに対して自サーバの削除を依頼し、管理を引き継ぐ管理サーバは、前記ドキュメントの管理情報を取得した後、その管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知することを特徴とするドキュメント分散処理システムのサーバ管理方法。

【請求項9】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおいて、それぞれ別々にオンラインドキュメントを共有する複数のシステムを一つに統合するために、第1のシステムにおけるドキュメントの管理情報を管理する管理サーバを第2のシステムにおける管理サーバに変更するにあたって、前記第1のシステムにおける管理サーバは、

自システムのドキュメントの管理情報を、前記第2のシステムにおける管理サーバへ配送するとともに、そのドキュメントの管理情報に対する他のサーバからの要求を、前記第2のシステムにおける管理サーバに対して転送し、前記第2のシステムにおける管理サーバは、前記ドキュメントの管理情報を取得した後、その管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知することを特徴とするドキュメント分散処理システムのサーバ管理方法。

【請求項10】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおけるサーバが用いるプログラムを記録した記録媒体であって、同一のドキュメントの複製を各サーバに配布する処理と、あるドキュメントの更新を開始する場合に、他のサーバに対して更新開始通知を送り、更新が終了したときに更新終了通知を送る処理と、他のサーバから更新開始通知を受け取ったときに該当するドキュメントに対する更新をロックし、更新終了通知を受け取ったときに前記ロックを解除する処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【請求項11】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおけるサーバが用いるプログラムを記録した記録媒体であって、同一のドキュメントの複製を各サーバに配布する処理と、あるドキュメントを更新する場合に、そのドキュメントの更新開始と同時に他のサーバに対して更新開始通知を送る処理と、更新開始通知に対する前記他のサーバからの承認の応答を受け取る前に、そのドキュメントの更新を開始する処理と、他のサーバからの更新の承認の有無を、更新が終了するまでの間に判定する処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【請求項12】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおけるサーバが用いるプログラムを記録した記録媒体であって、ドキュメントをその内容である実体と、その実体にリンクするメタ情報を持つ論理オブジェクトとに分離し、ドキュメントを作成または更新した場合に、他のサーバに対して論理オブジェクトの複製のみを送信する処理と、その後、他のサーバからの請求または所定の契機で、前記作成または更新された実体の複製を送信する処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【請求項13】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムに

おけるサーバが用いるプログラムを記録した記録媒体であって、各サーバに設定された優先度を管理するとともに、各サーバごとに前記優先度に対応して設定された独自のドキュメント版体系を管理する処理と、ドキュメントを更新する場合に、最も優先度の高いサーバに対し改版申請を送信し、改版が承認された場合にその版を正式版とする処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【請求項14】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおけるサーバが用いるプログラムを記録した記録媒体であって、各サーバに設定された優先度を管理するとともに、各サーバごとに前記優先度に対応して設定された独自のドキュメント版体系を管理する処理と、ドキュメントを更新する場合に、前記優先度が最も高いサーバに対して改版申請を送信し、前記優先度が最も高いサーバとの通信が途絶している場合には、前記通信が途絶したサーバの次に高い優先度を持つサーバに対して改版申請を送信し、改版が承認された版を正式版とする処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【請求項15】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおけるサーバが用いるプログラムを記録した記録媒体であって、システムへの新規サーバの登録依頼があった場合に、依頼された時点における共有ドキュメントの複製を、新規サーバへ配送する処理と、その後に更新されたドキュメントのデータの新規サーバへの配送を、全サーバへの新規サーバの登録が完了するまで代行する処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【請求項16】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおけるサーバが用いるプログラムを記録した記録媒体であって、あるドキュメントの管理情報を管理する機能を他のサーバに移行するとき、移行するドキュメントの管理情報を移行先の管理サーバへ配送する処理と、そのドキュメントの管理情報に対する他のサーバからの要求を、移行先の管理サーバに対して転送する処理と、自サーバが移行先のサーバとなると、移行元のサーバからドキュメントの管理情報を取得した後、その管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知する処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【請求項17】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムに

おけるサーバが用いるプログラムを記録した記録媒体であって、自サーバをシステムから削除する場合に、自サーバが管理するドキュメントの管理情報を、管理を引き継ぐサーバに対して配送する処理と、そのドキュメントの管理情報に対する他のサーバからの要求を、前記管理を引き継ぐ管理サーバに対して転送する処理と、前記管理を引き継ぐ管理サーバからの引き継ぎの確認後に他のサーバに対して自サーバの削除を依頼する処理と、自サーバが他のサーバから、あるドキュメントの管理情報の管理を引き継ぐ場合に、前記ドキュメントの管理情報を取得した後、その管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知する処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【請求項18】 ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするシステムにおけるサーバが用いるプログラムを記録した記録媒体であって、それぞれ別々にオンラインドキュメントを共有する複数のシステムを一つに統合するために、自サーバが管理する自システムにおけるドキュメントの管理情報の管理機能を、他のシステムにおける管理サーバに変更するにあたって、自システムのドキュメントの管理情報を、前記第2のシステムにおける管理サーバへ配送する処理と、そのドキュメントの管理情報に対する他のサーバからの要求を、前記第2のシステムにおける管理サーバに対して転送する処理と、自サーバが他のシステムの管理サーバからドキュメントの管理情報の管理機能を引き継ぐ場合に、そのドキュメントの管理情報を取得した後、その管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知する処理とを、計算機に実行させるプログラムを記録したことを特徴とするサーバプログラムの記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 複数拠点（サーバ）における事務系、開発系の業務などで、各拠点間で同一のドキュメント（文書）を互いに参照・更新することがある。ドキュメントがオンライン上のファイルであるような場合には、単一の文書を転送するか、同一内容の文書を複数のサイトで共有する方法がある。

【0002】 本発明は、同一内容の文書を複数のサイトで共有する方式に係り、特に、複数拠点で分散されたオンラインドキュメントを使用する際に、必要最小限のデータ通信を利用し、ドキュメントの同時共有および任意の拠点における更新を可能としたドキュメント分散処理方法、およびマルチサーバ環境においてサーバを追加登録、または変更する場合のドキュメント分散処理システムのサーバ管理方法に関する。

【0003】

【従来の技術】 従来、複数拠点間で文書を共有する技術として、以下の方法がある。

（1）ワークフロー等を利用し、単一の文書を拠点間で転送する方法。

【0004】 （2）同一文書の複製を拠点間で保有し、複製された文書に対し参照・更新を行う方法。

後者の（2）に示す同一文書の複製を拠点間で保有する方法の場合には、複数の拠点で同一文書に関する更新が発生し、内容の異なる文書が複数存在する可能性があるが、この対処方法として、以下の方法がある。

【0005】 （2-a）更新可能な拠点を1箇所限定し、他の拠点では更新を禁止する方法。

（2-b）複数の拠点での更新を許可し、複製を統合する際に何らかの手段で文書間の差異の修正を行う方法。

【0006】 また、従来のマルチサーバによる資源管理の技術は、マスタ・スレーブ方式のような集中型のものがほとんどであった。マスタ・スレーブ方式のシステムに新しいサーバを登録する場合には、マスタに相当するサーバに登録を依頼し、依頼されたマスタサーバは、新しいサーバに資源を配布できるようにするための変更を行い、トランザクション中の制御も1箇所で行っていた。

【0007】

【発明が解決しようとする課題】 従来の（1）に示す方法は、文書を更新する手順が拠点間で予め決定されている場合には問題がないが、複数拠点で任意に参照・更新を行う必要があるような非定型業務の場合には不適である。

【0008】 また、（2-a）に示す方法は、更新を行える拠点が限定されているため、やはり自由に更新を行うことができないという問題がある。（2-b）に示す方法は、多重更新が頻繁に発生してしまうことが問題であり、多重更新により生じてしまった差異を解決する手段として一般に有効な方法がない。

【0009】 また、従来のマスタ・スレーブ方式のシステムの場合には、新しいサーバの追加、既存のサーバの削除などが必要となったとき、特定の1箇所でのサーバの登録変更が行われるため、管理情報の一貫性の保持についての問題は発生しなかった。しかし、ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするようなドキュメント分散処理システムは、マスタに相当するデータやトランザクションを一括管理するサーバは存在せず、すべてのサーバが対等に存在し、自由にサーバの追加変更が行えるマルチサーバ環境であることが望ましい。

【0010】 このような各サーバが対等なマルチサーバ環境では、任意の箇所でのサーバ変更が発生することがあるために、サーバごとにサーバの登録状態の不整合が発生するという問題がある。サーバ登録の不整合がある

と、資源に関する更新処理が失敗したり、生成されたドキュメントが届かないことが起こる。これを回避するためには、ドキュメントを参照、更新するシステムのサービスを一旦止めて、全サーバで同期的にサーバを追加・削除するなどの必要があり、その間におけるサービス性が著しく阻害される。

【0011】本発明は上記問題点の解決を図り、サービス性のよいドキュメント分散処理システムを実現することを目的とする。特に、本発明は、複数の拠点における任意の更新を許容するが、1箇所の拠点で更新を行っている間のみ、他の拠点では更新を許可しない方式によって、多重更新の発生頻度を減少させ、かつ更新の際のロック要求に関する待機時間を短縮するような方式を提供することを目的とする。また、本発明は、各サーバが対等なマルチサーバ環境でも、同期のためのサービスの中断によって、他の処理が待たされることがないようにすること、かつ、一貫性を保持しながらサーバの追加、変更を可能にすることを目的とする。

【0012】

【課題を解決するための手段】図1は、本発明のドキュメント分散処理方法の原理説明図である。図1において、ネットワーク40で結ばれた複数の拠点には、それぞれ、サーバ10a、10bと、レポジトリ20a、20b、サーバに接続するクライアント30a、30bが存在する。各拠点はサーバ・クライアント構成になっている。

【0013】メッセージ処理手段1a、1bは、他サーバへ送信するメッセージの作成、および他サーバから到着したメッセージの処理を行う手段である。本システムでは、ドキュメントの論理オブジェクト（ドキュメントのメタ情報）および実体もメッセージに変換して配送するので、メッセージ処理手段1a、1bは、配送の際にオブジェクト／ファイルからメッセージへの変換および配送されてきたメッセージからオブジェクト／ファイルへの復元も担当する。

【0014】メッセージ通信手段2a、2bは、他サーバへのメッセージ送信および他サーバからのメッセージ受信を担当する手段である。レポジトリ管理手段3a、3bは、レポジトリ20a、20bに格納されている論理オブジェクトや実体の参照、検索、更新等を行う。

【0015】版数管理手段4a、4bは、ドキュメントの版管理を担当する。ロック管理手段5a、5bは、ドキュメントのロック、ロック解除、ロック状態の管理を担当する。

【0016】時刻監視手段6a、6bは、メッセージ通信のタイムアウト発生および定期レプリケーション時刻の通知を、メッセージ処理手段1a、1bに対して行う。ユーザインタフェース（UI）7a、7bは、利用者からのドキュメントの参照、更新操作を受け付ける手段である。

【0017】レポジトリ20a、20bは、ドキュメントの実体ファイルや論理オブジェクトを格納する記憶手段である。例えば、データベースに類似する構造を持つ格納領域を持つ。

【0018】クライアント30a、30bは、文書を編集・参照するドキュメントアプリケーションを持つ。本発明では、同一ドキュメント（文書）が各サーバに配置される。あるサーバ10aで生成・更新されたドキュメントの内容は、ネットワーク40を介して他のサーバ10bが持つドキュメントに反映される。

【0019】本発明は、以下のように作用する。

(1) サーバ10aは、あるドキュメントの更新を開始すると、メッセージ通信手段2aによりサーバ10bに対して更新開始通知を送る。サーバ10bは、メッセージ通信手段2bにより更新開始通知を受け取り、ロック管理手段5bにより該当するドキュメントに対する更新をロックする。

【0020】サーバ10aは、更新が終了したときにメッセージ通信手段2bにより更新終了通知を送り、サーバ10bは、ロック管理手段5bにより更新終了通知を受け取ったときに前記ロックを解除する。

【0021】これにより同一ドキュメントに対する多重更新を回避することができる。

(2) サーバ10aは、あるドキュメントの更新を開始すると同時にサーバ10bに対して更新開始通知を送り、ユーザインタフェース（UI）7aによりロック完了通知を受け取る前に自クライアントに対して更新開始を承認し、更新を進め、他のサーバ10bからのロック完了通知による更新の承認の有無は、更新が終了するまでの間に判定する。

【0022】これにより、ロック完了通知の受信を待たずに更新を開始することができ、実際の処理開始までの時間の短縮ができる。

(3) レポジトリ20aでは、ドキュメントの実体ファイルとこれにリンクするドキュメントのメタ情報である論理オブジェクトとを分離して持ち、サーバ10aはドキュメントを作成または更新する際に、論理オブジェクトの複製のみを送信し、その後、他のサーバ10bからの請求または所定の時間間隔により、作成または更新されたドキュメントの実体の複製を送信する。

【0023】この論理オブジェクトは、最新の实体を持つサーバの情報を含む。サーバ10aがドキュメントの实体とその論理オブジェクトを持ち、サーバ10bが論理オブジェクトだけを持つ状態で、サーバ10bがドキュメントを参照または更新する場合、サーバ10bは、論理オブジェクトをもとに最新の实体を保持するサーバ10aから最新の实体を取得する。

【0024】これにより、更新ごとに発生する实体の送受信のための通信量を軽減することができる。

(4) 各サーバ10a、10bに優先度を設定し、ド

キュメントの更新を開始するサーバ10aは、自サーバの優先度を付加した更新開始通知を、他のすべてのサーバ10bへ送信する。更新開始通知を受信したサーバ10bにおいて、指定されたドキュメントがロック状態である場合、そのサーバ10bでは、現在のロック要求に係るサーバの優先度と、受信したサーバ10aの優先度とを比べて、最終的に優先度の高いサーバによるロックのみを実施する。

【0025】これにより、複数のサーバで同時に更新を開始した場合に、更新の競合を回避することができる。

(5) 各サーバごとに優先度に対応する独自のドキュメント版体系を設定し、ドキュメントを更新するサーバ(例えばサーバ10a)は、最も優先度の高いサーバ(例えばサーバ10b)に対し改版申請を送信し、改版が承認された場合には、承認された版を持つドキュメントを正式版として採用する。その際に必要であれば、その正式版のドキュメントを他のサーバへ送信する。

【0026】これにより、複数のサーバで改版が発生した場合でも、同一の版で異なる内容のドキュメントが存在してしまうという事態を防止し、版の分裂を解決することが可能となる。

【0027】(6) サーバの優先度が最も高いサーバとの通信が途絶している場合には、通信が途絶したサーバの次に高い優先度を持つサーバに対して改版申請を送信する。

【0028】これにより、いずれかのサーバとの通信が途絶した状態の場合であっても、残りのサーバ間で同一のドキュメントの版管理を行うことができる。

(7) また、各ドキュメント単位にサーバの優先度を設定する。

【0029】これにより、ドキュメントごとに優先度の高いサーバが異なり、改版の承認に関する処理が分散されるので、通信の集中を回避することができる。図2は、本発明によるサーバ管理のための構成を説明する図である。

【0030】図1に示す各サーバ10a、10b、…のメッセージ処理手段1は、サーバの登録、削除、ドキュメントの管理情報を管理するサーバの変更、複数のシステムの統合のために、図2に示すように、サーバ登録手段51、サーバ削除手段52、管理サーバ変更手段53、システム統合手段54を持つ。サーバ一覧テーブル55は、システムに登録されているサーバの情報を保存するテーブルである。

【0031】サーバ登録手段51は、システムへの新規サーバの登録依頼があった場合に、依頼された時点における共有ドキュメントの複製を、新規サーバへ配送する処理を行うとともに、その後に更新されたドキュメントのデータの新規サーバへの配送を、全サーバへの新規サーバの登録が完了するまで代行する処理を行う。

【0032】また、サーバ削除手段52は、自サーバを

システムから削除する場合に、自サーバのレポジトリ管理手段3が管理するドキュメントの管理情報を、管理を引き継ぐサーバに対して配送する処理と、そのドキュメントの管理情報に対する他のサーバからの要求を、管理を引き継ぐ管理サーバに対して転送する処理と、管理を引き継ぐ管理サーバからの引き継ぎの確認後に他のサーバに対して自サーバの削除を依頼する処理とを行う。また、サーバ削除手段52は、自サーバが他のサーバからドキュメントの管理情報の管理を引き継ぐ場合に、そのドキュメントの管理情報を取得した後、レポジトリ管理手段3によりその管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知する処理を行う。

【0033】管理サーバ変更手段53は、あるドキュメントの管理情報を管理する機能を他のサーバに移行するとき、移行するドキュメントの管理情報を移行先の管理サーバへ配送する処理と、そのドキュメントの管理情報に対する他のサーバからの要求を、移行先の管理サーバに対して転送する処理とを行う。さらに、自サーバが管理移行先のサーバとなるときには、移行元のサーバからドキュメントの管理情報を取得した後、レポジトリ管理手段3によりその管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知する処理を行う。

【0034】システム統合手段54は、それぞれ別々にオンラインドキュメントを共有する複数のシステムを一つに統合するために、自サーバのレポジトリ管理手段3が管理する自システムにおけるドキュメントの管理情報の管理機能を、他のシステムにおける管理サーバに変更するにあたって、自システムのドキュメントの管理情報を、他のシステムにおける管理サーバへ配送する処理と、そのドキュメントの管理情報に対する他のサーバからの要求を、前記他のシステムにおける管理サーバに対して転送する処理とを行う。さらに、自サーバが他のシステムの管理サーバからドキュメントの管理情報の管理機能を引き継ぐ場合には、そのドキュメントの管理情報を取得した後、その管理情報の管理を開始し、他のサーバに対して管理サーバの移行を通知する処理を行う。

【0035】以上の各処理手段をサーバの計算機によって実現するためのプログラムは、計算機が読み取り可能な可搬媒体メモリ、半導体メモリ、ハードディスクなどの適当な記録媒体に格納することができる。

【0036】

【発明の実施の形態】以下に、本発明の実施の形態を説明する。はじめに、本発明のドキュメント分散処理方法の実施の形態を説明する。

【0037】本実施の形態を説明する前提として、遠隔の拠点間で文書を相互に共有するような業務を行う場合を例とする。各拠点には、コンピュータと、ファイルや論理オブジェクトを格納可能なレポジトリ(データベース)と、サーバアプリケーションが存在する。レポジトリは、文書(ドキュメント)を版(バージョン)管理

し、版ごとに文書を格納できる機能を持つ。また、拠点はすべてネットワークに接続されており、任意の拠点間でメッセージ通信、ファイルや論理オブジェクトの転送が可能である。各拠点は、サーバ・クライアント構成になっており、クライアントには文書を編集・参照するドキュメントアプリケーションが実装されている。

【0038】図3は、このようなシステムにおける文書の複製と配布を説明する図である。図3に示すように、同一文書を複製したものを各サーバA、B、Cに配置する。サーバAまたはサーバAに接続されているクライアントにより文書1を生成または更新した場合には、サーバAは、ネットワークで結ばれた他のサーバB、Cに対し文書1の複製を送信する。

【0039】図4は、送信処理および受信処理の処理フローチャートである。送信処理では、図4(A)に示すように、文書を作成・更新したサーバは、文書を複製し(S101)、その複製を他のサーバに送信する(S102)。受信処理では、図4(B)に示すように、サーバは複製を受信し(S103)、複製を自己の文書に反映させる(S104)。なお後述するが、文書の複製は、文書の実体を複製する以外に、文書に関するメタ情報である論理オブジェクトのみを複製してもよい。

【0040】〔第1の処理方式〕第1の処理方式では、サーバは更新開始時に他のサーバに対して、更新開始を通知する。他のサーバは更新開始通知を受けとった時点でその文書をロックする。ロックされた文書は、ロックが解除されるまでそのサーバでは更新不可能になる。更新を完了したサーバは、他のサーバに更新後の文書を配布する。他のサーバは、配布された文書を受け取った時点でロックを解除し、更新結果を自己の文書に反映させる。

【0041】図5は、第1の処理方式の手順の例を示す図である。サーバAは、クライアントからの更新開始を契機に、他のサーバB、Cに対して、ロック要求を含む更新開始通知を送信する。他のサーバB、Cは、更新開始通知を受信した時点でその文書をロックし、ロック完了通知をサーバAに返信する。

【0042】その後、更新を終了したサーバAはその文書のロックを解除し、他のサーバB、Cに対して更新後の文書(更新文書)を配布する。他のサーバB、Cは、更新文書を受信した時点で文書のロックを解除し、自サーバの文書に更新結果を反映させる。

【0043】図6は、第1の処理方式における処理フローチャートである。図6(A)に示す送信処理においては、クライアントが文書更新の開始を宣言すると(S110)、他のサーバに対してロック要求を含む更新開始通知を送信する(S111)。他のサーバからのロック完了通知を受信し(S112)、その後文書を更新し(S113)、その文書の論理オブジェクトまたは実体の複製を他のサーバへ送信する(S114)。

【0044】図6(B)に示す受信処理においては、文書更新を開始したサーバからロック要求を含む更新開始通知を受信すると(S115)、文書をロックし(S116)、直ちにロック完了通知を送信する(S117)。続いて、文書の論理オブジェクトまたは実体の複製を受信し(S118)、文書のロックを解除する(S119)。

【0045】〔第2の処理方式〕上記第1の処理方式では、他のすべてのサーバB、Cでロックが行われ、その結果として返される結果(ロック完了通知)を確認した後に、更新開始の許可を利用者(クライアント)に与えているため、通信に時間を要するような環境では、利用者が更新開始を宣言してから実際に更新処理を始めるまで、かなりの時間待たされることになる。

【0046】第2の処理方式では、この点を改良し、文書を更新するサーバは、他サーバからのロック完了通知の受信を待たずに、自サーバのクライアントに更新を許可し、ロックの正否(他サーバからの更新承認)の判定は更新終了まで延期する。これにより、クライアントは、更新を先行的に実行することができ、時間の短縮が可能になる。

【0047】図7は、第2の処理方式の手順の例を示す図である。サーバAは、クライアントから更新開始の依頼を受けると、サーバBおよびサーバCに対しロック要求を含む更新開始通知を送信した後、直ちにクライアントに更新開始OKを出して更新を許可する。サーバB、Cは、ロック要求を含む更新開始通知を受信して、該当する文書をロックし、ロック完了通知をサーバAに返す。

【0048】サーバAでは、更新終了までに他サーバからのロック完了通知を受信し、他サーバB、Cによる更新承認の判定を行い、更新を終了する。もし、更新終了までの間に、他サーバB、Cからロック完了通知を受信しなかった場合には、クライアントによる文書の更新を無効にして文書を更新開始前の状態に戻すか、またはその文書をローカル版として保存し、処理を終了する。

【0049】図8は、第2の処理方式における処理フローチャートである。図8(A)に示す送信処理においては、クライアントが文書更新の開始を宣言すると(S120)、他のサーバに対してロック要求を送信して(S121)、直ちに文書更新を許可する(S122)。更新が終了するまでの間に、他のサーバからのロック完了通知を受信し、他のすべてのサーバからロック完了通知を受信したなら(S123)、更新された文書の実体または論理オブジェクトの複製を他のサーバに送信する(S124)。

【0050】図8(B)に示す受信処理においては、文書更新を開始したサーバからロック要求を受信し(S125)、その文書をロックし(S126)、ロック完了通知を要求元のサーバへ送信する(S127)。文書を

更新したサーバから、文書の実体または論理オブジェクトの複製を受信したなら（S128）、それを自サーバ内の文書に反映し、ロックを解除する（S129）。

【0051】〔第3の処理方式〕文書を共有するシステムにおいて、いずれかのサーバで文書が新規作成・更新された場合には、更新内容を他のサーバに反映させる必要があるが、更新のたびに文書内容の転送を行う方法では、ファイルサイズが大きい場合やサーバ間の通信に時間がかかる場合には通信の負荷が問題となる。

【0052】第3の処理方式では、各サーバにおける文書を、文書の実体である文書ファイル（以下、実体ファイルという）そのものと、文書に関する情報（実体ファイルへのリンク、文書名、作成者、作成日付、版数等）のみで構成される論理オブジェクトとの二つに分離して実装する。図9は、文書の実装形態の例を示す図である。

【0053】クライアントから参照・更新のために文書を利用する場合には、論理オブジェクトを経由して実体ファイルの取得を行う。また、文書を論理オブジェクトと実体ファイルとに分離して実装することで、文書の新規作成時や更新時には、文書実体を配布せずに論理オブジェクトのみを配布し、実体ファイルは、通常更新が発生しない時間帯や通信負荷が小さい時間帯などを指定しておき、定期的に配布する。

【0054】図10は、第3の処理方式を説明する図である。サーバAで文書を新規作成または更新すると、図10（A）に示すように、他のサーバB、Cに対しては、更新後の論理オブジェクト（新）のみを複製して即時に配布する。更新後の文書の実体ファイル（新）は、直ちに送ることはしないで、図10（B）に示すように、通常、更新が発生しない時間帯などに定期的に複製して配布を行う。

【0055】さらに、論理オブジェクトに、最新実体の保有の有無、最新実体を保有するサーバの識別子の情報を追加することもできる。図11は、最新の実体の取得処理の処理フローチャートである。

【0056】利用者が更新を開始するなど、最新の文書の実体が必要な場合には、自サーバの持つ論理オブジェクトに対し、最新実体の有無を問い合わせる（S130、S131）。自サーバが最新の实体ファイルを保有していない場合には、論理オブジェクトより最新の实体ファイルを保有するサーバの識別子を得て、そのサーバに最新の实体ファイルを要求し、必要な最新の实体ファイルを転送してもらい（S132）、文書処理を行う（S133）。自サーバが最新の实体ファイルを保有している場合には、その実体ファイルを使用して文書処理を行う（S133）。

【0057】〔第4の処理方式〕複数のサーバが同時に更新を開始した場合、互いに文書のロックができないため、個々のサーバでそれぞれ更新が行われてしまう。そ

こで、同一文書に対するロック要求が複数のサーバから発生した場合には、いずれかのロックを有効にするために、各サーバにあらかじめ異なる優先度を設定し、文書をロックする際に、ロックを要求した（更新の発生した）サーバの優先度を同時に記録する。このため、他のサーバに対してロックを要求する場合には、優先度情報を付加して送信する。

【0058】ロックを要求されたサーバでは、既にその文書がロック状態である場合、ロックを要求したサーバの優先度を比較する。もし、新しくロックを要求したサーバの優先度がロック中のサーバの優先度よりも高い場合には、現在のロックをキャンセルし、新たに到着したロック要求を実行する。新しくロックを要求したサーバの優先度がロック中のサーバの優先度よりも低い場合には、現在のロックを優先し、新たに到着したロック要求を拒絶する。

【0059】図12は、第4の処理方式を説明する図である。サーバA、B、Cにおいて、サーバAの優先度がサーバBの優先度より高いとする。図12（A）に示すように、サーバCにおいてサーバBからの依頼による文書ロック状態である場合に、サーバAからのロック要求が到着したときには、サーバAの優先度がサーバBの優先度より高いため、サーバCは、現在のサーバBによる文書ロックをキャンセルし、到着したサーバAによるロック要求を実行する。

【0060】一方、図12（B）に示すように、サーバCにおいてサーバAからの依頼による文書ロック状態である場合に、サーバBからのロック要求が到着したときには、既に優先度の高いサーバAによる文書ロックがされているのでこれを優先し、サーバCは、サーバBからのロック要求を却下する。

【0061】図13は、第4の処理方式における処理フローチャートである。あるサーバが、文書ロック状態であり（S140）、他のサーバからロック要求を受信したなら（S141）、受信したロック要求の要求元のサーバの優先度と、現在のロックを要求したサーバの優先度とを比較し（S142）、要求元サーバの優先度が現ロックの要求元サーバの優先度より高い場合には、現在の文書ロックをキャンセルし、受信したサーバのロック要求を実行する（S143）。受信したサーバの優先度が現ロックの要求元のサーバの優先度より高くない場合には、受信したロック要求を却下する（S144）。

【0062】〔第5の処理方式〕前述した処理方式では、先行的に更新を行うことを許容しているため、更新に要する時間が短い等の原因で、図14に示すように多重更新が発生してしまう可能性がある。この場合、同じ版に更新してしまうと、同一の版で異なる内容の文書が存在してしまうという問題が発生する。

【0063】そこで、各サーバ単位に独自の版体系を設定し、配布された版から分枝できるようにする。そし

て、サーバの優先度に対応して版体系に優先度を設定し、このうち最も優先度の高い版体系をそのシステムにおける正式版とする。

【0064】図15は、第5の処理方式の手順の例を示す図である。まず、サーバBは、文書の更新時に、他のサーバであるサーバCに対してロック要求を送信するとともに、そのシステムで最も優先度の高いサーバAに対して正式版の改版申請を送信する。同様に、サーバCも文書を更新するときには、他のサーバBにロック要求を送信し、サーバAに対して改版申請を送信する。

【0065】サーバAでは、改版申請された文書がロックされていなければ、先に到着したサーバBの改版申請に対して改版承認のメッセージを返送する。改版承認を既に発行した後に届いたサーバCの改版申請に対しては、改版却下のメッセージを返送する。

【0066】改版承認を受け取ったサーバBは、他のサーバA、Cに対して承認された新版の実体または論理オブジェクト（以下、文書オブジェクトという）を複製し配布することにより、改版を行う。これにより、サーバBにおいて更新された文書1の正式な版数は「A-2」となる。

【0067】サーバAでは、サーバBから配布された新版文書オブジェクト（文書1、版：A-2）を正式版とする。また、改版申請を却下されたサーバCでは、文書更新が取り消され、サーバBから配布された新版文書オブジェクト（文書1、版：A-2）を正式版とする。ここで、サーバC内で文書1（版：C-2）の更新が完了していた場合には、この文書1（版：C-2）は、サーバCのローカル版文書として保存される。

【0068】図16は、第5の処理方式の処理フローチャートである。更新サーバにおいては、図16（A）に示すように、文書更新を開始すると（S150）、正式版を発行するサーバに対して改版申請を送信して（S151）、文書を更新する（S152）。まず、自サーバにおける独自の版体系で改版し（S153）、正式版を発行するサーバからの改版承認を受信するまで待機し（S154）、改版承認を受信したなら、承認された正式版で改版を行う（S155）。

【0069】正式版を発行するサーバにおいては、図16（B）に示すように、文書更新を開始したサーバから改版申請を受信すると（S156）、既に他のサーバに改版承認を送信したかどうかを判断し（S157）、既に改版承認を送信している場合には、新しい申請元のサーバに対し改版却下を送信し（S158）、未だ改版承認を送信していない場合には、改版承認を送信する（S159）。

【0070】〔第6の処理方式〕上述した第5の処理方式では、仮に最も優先度の高いサーバとの通信が途絶している場合に、ローカル版以外の改版が行えなくなってしまうという問題が生じる。そこで、改版申請を送信し

てから応答を受信するために待つ時間の限界（タイムアウト）を設定し、タイムアウトが発生した場合に、次に優先度の高いサーバに対して改版申請を再送信し、そのサーバにより改版承認が得られたときに、得られた版の文書オブジェクト（実体または論理オブジェクト）を複製し配布する。

【0071】図17は、通信途絶が発生した場合の手順の例を示す図である。サーバCが更新時に、システム内で優先度が最も高いサーバAに対して改版申請を送信したにもかかわらず、そのサーバAもしくは通信路の障害等によりその通信が途絶した場合には、サーバCは、改版申請の応答に対するタイムアウトを検出し、次に優先度の高いサーバBに対して改版申請を再送信する。サーバBがサーバCに対して改版承認を発行すると、サーバCは、更新を完了した新版文書オブジェクト（版：C-2）を複製し、サーバBに配布して改版を行う。

【0072】図18は、第6の処理方式によるタイムアウトが発生した場合の処理フローチャートである。更新サーバにおいて、文書更新を開始すると（S160）、正式版を発行するサーバに対して改版申請を送信して

（S161）、文書を更新する（S162）。自サーバにおける独自の版体系で改版し（S163）、正式版を発行するサーバからの改版承認を受信したかどうかを判断する（S164）。改版承認を受信したなら、承認された正式版で改版を行う（S165）。改版承認を受信しないときにはタイムアウト発生まで待機して（S166）、タイムアウト発生であれば、次の優先度のサーバに対し改版申請を送信して（S167）、S164以下の処理を行う。なお、改版申請が却下された場合には、その文書を自サーバのみのローカル版文書として扱うことは、前述した第5の処理方式と同様である。

【0073】〔第7の処理方式〕上述した第5および第6の処理方式においては、最も優先度の高いサーバに通信が集中する傾向がある。そこで、本方式では、負荷分散のために、各文書単位にサーバの優先度を設定する。これにより、文書ごとに最も優先度の高いサーバが分散されるので、通信の集中を回避することができる。

【0074】次に、遠隔地の3拠点において、文書を互いに共有するような業務に本発明を適用した実施例を説明する。ネットワークで結ばれた拠点A、B、Cでは、比較的大きなサイズの文書を扱い、文書の更新が行われる主な拠点は、文書の種類によってほぼ決まっているものとする。例えば「設計書」という種類の文書は、主に拠点Aで更新され、拠点B、Cでも更新はされるが頻度は少ないものとする。

【0075】文書は、文書内容が格納された実体ファイルと論理オブジェクトとに分離して各拠点のリポジトリに格納される。各拠点には、サーバ、クライアントの装置が設置され、利用者はクライアントから各拠点にあるサーバに接続して作業を行う。クライアントからは文書

の作成・登録、参照および編集が可能である。

【0076】サーバAで「設計書」である文書1を新規作成した場合、サーバBおよびサーバCに対して文書1の論理オブジェクトが配布される。したがって、サーバB、Cの利用者は、クライアント上で新規生成された文書1の情報（文書名、作成者、作成日時、バージョン等）を論理オブジェクトから取得することができる。

【0077】ここで、文書の実体（ファイル）はサーバBおよびサーバC上には送られていない。サーバB上で文書1を参照したい場合、利用者はクライアントから文書1の実体を要求する。そうすると、サーバBは、サーバAに対して実体の配布を要求し、サーバAからサーバBへ実体のコピーが行われる。コピーが完了するとクライアントを介して利用者に通知が行われる。

【0078】また、その日に作成された文書1は、1日1回、作業の少ない時間帯を指定して、すべてのサーバに対して一括して実体（ファイル）を配布するように設定する。したがって、前日に作成された文書は翌日以降にはどの拠点においても、配布要求を行わずに、直ちに取得することができる。

【0079】「設計書」はサーバAで最も多く更新が行われるので、更新の優先度をサーバAが最も高く、次いでサーバB、サーバCの順とする。サーバAは最も優先度が高いので、サーバB、サーバCの利用者が先に更新を開始していても、後からサーバAで同じ文書の更新が開始されたなら、サーバB、Cの利用者は、更新のキャンセルを要求される場合がある。

【0080】また、サーバB、サーバCでは同じ1.0版から更新を行った場合、更新の終了直後ではサーバBにおいては1.1/B版、サーバCにおいては1.1/C版のようなそのサーバ固有の版（ローカル版）が生成される。その後、サーバAからの2.0版への改版許可が得られた時点で、1.1/Bのようなローカル版は、グローバル版である2.0版に置き換えられる。なお、ローカル版はそのサーバにおいて保存される。

【0081】次に、本発明のドキュメント分散処理システムのサーバ管理方法の実施の形態を説明する。すべてのサーバが対等に存在し、自由にサーバの追加変更を行えるマルチサーバ環境を実現するために、以下に説明する管理方式を用いる。

【0082】図19は、サーバ一覧テーブルの構成例を示す図である。サーバ一覧テーブルには、サーバ名、宛て先（アドレス）、サーバの属性に関する情報を記憶する。サーバの属性には、例えばそのサーバがどのような管理情報を管理しているかなどの情報が含まれる。このサーバ一覧テーブルは、全サーバが同じ内容のものを保持し、各サーバ間の通信に用いられる。

【0083】〔第1の管理方式〕各サーバがデータ（ドキュメント）の複製を共有し、任意のサーバでデータ更新が行えるようなマルチサーバシステムにおいて、サー

バを登録する場合を考える。あるサーバで更新されたデータは、その複製が別のサーバに配送される。

【0084】もし、データが特定のサーバだけで更新されるのであれば、新規登録のサーバは、データの更新を管理するサーバからデータを取得すればよい。しかし、データが任意の箇所で更新される可能性があるため、あるサーバからデータを取得している間に他のサーバがデータを更新した場合には、登録した時にはその新しい更新データを取得することができず、その結果、サーバ間で所有するデータに差異が生じてしまうおそれがある。

【0085】本発明では、サーバの登録が完了し、新規サーバが他の全サーバからデータを受け取ることが確認されるまで、任意の登録されている1サーバを指定し、指定されたサーバが新規サーバへのデータの配送を代行するようにする。この手順を図20に示す。

【0086】図20は、第1の管理方式の手順の例を示す図である。

（P11）本システムへの追加登録を希望する新規サーバXは、すでにシステムに登録されているサーバA～Dの一つ（例えばサーバA）に、現時点で登録されているサーバの情報を持つサーバ一覧テーブルの送付と、現時点で共有されているデータの複製の配送および新たな更新データの配送代行を依頼する。

【0087】（P12）サーバAは、自己の持つサーバ一覧テーブルにサーバXを追加登録し、そのサーバ一覧テーブルとデータを新規サーバXに送信する。

（P13）以後、依頼されたサーバAは、サーバXから転送終了の依頼があるまで、他のサーバ（例えばサーバB）から自サーバに配送されてきたデータをすべて新規サーバXに転送する。

【0088】（P14）新規サーバXは、サーバ一覧テーブルをもとに、登録されているサーバB～D（またはサーバA～D）に対して、自サーバの登録を依頼する。

（P15）サーバB～Dは、新規サーバXからの登録依頼を受け取った場合、新規サーバXを自サーバが持つサーバ一覧テーブルに登録し、登録完了を新規サーバXに通知する。この登録以降、サーバB～Dは、新規サーバXに対して直接更新されたデータを配送する。

【0089】（P16）新規サーバXは、サーバB～Dから登録完了通知を受け取ると、最初に配送の代行を依頼したサーバAに対して、代行の終了を依頼する。代行を行っているサーバAは、代行終了（転送終了）依頼を受け取ったら、新規サーバXへの配送代行を終了する。

【0090】図21は、第1の管理方式における処理フローチャートである。新規登録を希望する新規サーバは、現時点で登録されているサーバ一覧を任意のサーバより取得し（ステップS201）、サーバ一覧に登録されているサーバの一つにデータ配送の代行を依頼する

(ステップS202)。代行を依頼されたサーバは、自分に配送されたデータをすべて新規サーバに転送する(ステップS203)。

【0091】新規サーバは、サーバ一覧をもとに、登録されたサーバすべてに対して自サーバの登録を依頼する(ステップS204)。他のサーバは、新規サーバからの登録依頼を受け取ったら、新規サーバを自己のサーバ一覧に追加登録し、登録完了を通知する。以降、他のサーバは、新規サーバに対して更新したデータを配送する(ステップS205)。

【0092】新規サーバは、すべてのサーバから登録完了通知を受け取ったら、最初に配送代行を依頼したサーバに対して、配送代行の終了を依頼する(ステップS206)。代行を依頼されたサーバは、代行終了(転送終了)依頼を受け取ったら、新規サーバへの配送代行を終了する(ステップS207)。

【0093】〔第2の管理方式〕マルチサーバ環境において、ある資源を管理するサーバを、他のサーバに変更する必要があることがある。ここで、ある資源の状態が、別のサーバによる資源の使用によって変更されるとすると、ある時点をもって一度に新しいサーバに資源管理を移行させなければならない。しかし、一般に、管理サーバの変更の認知には各サーバごとに時間的なずれが生じるので、新しい管理サーバへの処理要求と古い管理サーバに対する処理要求とが混在する時間帯が存在する。これらの要求は、一貫性を保持するためにどちらか一方のサーバで処理されることが必要になる。資源の状態とは、例えばドキュメントの版数などである。この問題を解決するための手順を図22に示す。

【0094】図22は、第2の管理方式の手順の例を示す図である。

(P21) 現在、ある資源の管理を管理サーバAが行っており、その管理を新管理サーバYに移行するものとする。新管理サーバYは、現在の管理サーバAに対して、資源の管理情報の配送と、自サーバへの処理の移行の依頼を申請する。

【0095】(P22) 管理サーバAは、新管理サーバYに対し、資源管理の承認を通知するとともに、資源の管理情報を送信する。

(P23) 以降、管理サーバAは、新管理サーバYから転送終了の依頼があるまで、サーバB～Dから管理サーバAに到着したその資源に対する処理要求を新管理サーバYに転送する。

【0096】(P24) 新管理サーバYは、資源管理情報を取得したなら、資源管理を開始し、サーバB～Dに対して管理サーバの移行を通知する。

(P25) サーバB～Dは、移行通知を受け取ったら、管理サーバAに対して確認通知を発行し、通知以降、その資源に関しては新管理サーバYに処理要求を発行するようにする。

【0097】(P26) 管理サーバAは、サーバB～Dから確認通知を受け取ったら、新管理サーバYへの処理要求の転送を終了する。図23は、第2の管理方式における処理フローチャートである。

【0098】ある資源に対する新管理サーバは、現在の管理サーバに対して、資源管理情報の取得と自サーバへの資源管理処理の移行を依頼する(ステップS211)。現在の管理サーバは、新管理サーバに対し、移行の承認と資源管理情報を送信する。以降、他のサーバから管理サーバにその資源に対する処理要求が到着した場合、その処理要求を新管理サーバに転送する(ステップS212)。

【0099】新管理サーバは、資源管理情報を取得したら、資源管理処理を開始し、他のサーバに対して管理サーバの移行を通知する(ステップS213)。他のサーバは、移行通知を受け取ったら、旧管理サーバに対して確認通知を発行する。以降は、その資源に関しては新管理サーバに処理要求を発行する(ステップS214)。旧管理サーバは、管理するすべてのサーバから確認通知を受け取ったら、新管理サーバへの処理要求の転送を終了する(ステップS215)。

【0100】〔第3の管理方式〕ある管理サーバをシステムから削除する場合、これまで削除する管理サーバが管理していた資源は別のサーバに移行させる必要がある。そのための第3の管理方式の手順では、第2の管理方式の手順により資源の管理を新たな管理サーバへ移行し、その後に旧管理サーバを削除することにより、システムを停止することなく管理サーバを削除できるようにする。この手順の例を、図22を用いて説明するが、(P21)～(P25)までの処理は、第2の管理方式の処理と同様であるので説明を省略する。

【0101】(P26') 管理サーバAは、サーバB～Dから管理サーバ移行の確認通知を受け取ったら、転送処理を終了し、サーバB～Dに対して自サーバの削除を依頼する。サーバB～D上において、サーバ一覧テーブルから旧管理サーバAが削除され、処理が完了する。

【0102】図24は、第3の管理方式における処理フローチャートである。図24のフローチャートのステップS221～S224までの処理は、図23の処理フローチャートのステップS211～S214と同様であるので、説明を省略する。

【0103】ステップS225で、旧管理サーバは、管理するすべてのサーバから確認通知を受け取ったら、転送処理を終了し、各サーバに対して自サーバの登録削除を依頼する。各サーバは、旧管理サーバの登録を削除し、処理を完了する(ステップS226)。

【0104】〔第4の管理方式〕複数のサーバがそれぞれ登録された二つのドキュメント分散処理システムを統合し、一つのシステムにする場合を考える。その際、資源に共通の管理サーバを変更する場合(例えば、各シス

テムで別々に設定していた管理サーバを一つにする場合)には、第2の管理方式と同様の移行処理が必要になる。図25にその手順を示す。

【0105】図25は、第4の管理方式の手順の例を示す図である。図25では、システムSとシステムTとを統合し、システムTにおいて管理サーバAが管理していた資源の管理を、管理サーバAから新管理サーバYに移行する場合の例を示している。

【0106】(P31) 新管理サーバYは、資源管理処理の移行を管理サーバAに依頼申請する。

(P32) 管理サーバAは、移行の承認を通知するとともに、資源管理情報および管理サーバAに登録されているシステムTのサーバ一覧テーブルを新管理サーバYに送信する。

【0107】(P33) 以降、管理サーバAは、その資源に対するサーバB、Cからの処理要求を新管理サーバYに転送する。

(P34) 新管理サーバYは、システムSおよびTを統合した新サーバ一覧テーブルおよび管理サーバの移行通知を、サーバB、C、Zに配送する。

【0108】(P35) サーバB、C、Zは、サーバ一覧テーブルと移行通知を受け取ったら、自サーバのサーバ一覧テーブルを更新し、管理サーバAに対して確認通知を発行する。以降その資源に関しては新管理サーバYに要求を発行するようにする。管理サーバAは、サーバB、C、Zから確認通知を受け取ったら、新管理サーバYへの処理要求の転送を終了する。

【0109】図26は、第4の管理方式における処理フローチャートである。新たに統合システムの管理サーバとなる管理サーバ(新管理サーバという)は、管理処理の移行を他のシステムの管理サーバ(旧管理サーバという)に依頼する(ステップS231)。旧管理サーバは、移行の承認と資源管理情報および旧管理サーバに登録されているサーバ一覧を新管理サーバに送信する。以降、旧管理サーバは、その資源に対する処理要求を新管理サーバに転送する(ステップS232)。

【0110】新管理サーバは、統合するシステムに登録されている新サーバ一覧および管理サーバの移行通知を各サーバに送信する(ステップS233)。各サーバは、新サーバ一覧と移行通知を受け取ったら、自サーバのサーバ一覧を更新し、旧管理サーバに対して確認通知を発行する。以降、その資源に関しては新管理サーバに処理要求を発行する(ステップS234)。旧管理サーバは、すべてのサーバから確認通知を受け取ったら、新管理サーバへの処理要求の転送を終了する(ステップS235)。

【0111】次に、文書を互いに共有するような業務に本発明を適用した実施例を説明する。まず、文書を互いに共有するような業務を行うシステムにサーバxを追加する場合を図27を用いて説明する。

【0112】初期状態では、サーバa、b、…、i、…、nは、それぞれ文書1(版:1)、文書2(版:1)、文書3(版:1)を持ち、サーバxは、未登録の時点で何も文書を持っていない。

【0113】はじめに、サーバxは、サーバaに対し、サーバ一覧(テーブル)と文書データを要求する。サーバaは、要求を受け取った時点で、サーバxへの複製文書の転送処理を開始し、サーバxにサーバ一覧とサーバaが持っている文書データを配布する。この間、サーバxは、他のサーバb、…、i、…、nに登録されていないので、サーバiで文書1を第2版に改版したとすると、文書1(2版)は、直接サーバxには届かないが、本発明では、サーバaは配布された文書1(2版)をサーバxに転送するので、サーバxは、これをサーバa経由で受け取ることができる。

【0114】サーバxは、サーバaからサーバ一覧と文書データを受け取ったら、サーバa、b、…、i、…、nに対してサーバxの登録を依頼する。サーバa、b、…、i、…、nにおいて、サーバxがサーバ一覧に登録されると、以後各自のサーバで改版した文書は、サーバxに対しても配送されるようになる。これ以降に、サーバiが、文書2を更新して2版にしたときには、サーバxは、文書2(2版)の複製を直接サーバiからの配送によって取得することができる。

【0115】サーバa、b、…、i、…、nは、サーバxに対して登録完了を通知する。サーバxは、すべてのサーバで登録完了されたことを確認したら、サーバaに対して転送処理の終了を依頼する。すべての処理完了後、サーバxを含めた各サーバは、文書1(1版、2版)、文書2(1版、2版)、文書3(1版)を所有しており、保有する文書の一貫性が保たれることになる。

【0116】次に、それぞれ自システムにおける文書を共有するシステムSとシステムTが独立して稼働している状態で、システムSとシステムTとを統合し、同時に統合後の全体の管理サーバをサーバSaに統一する場合の例を、図28を用いて説明する。

【0117】まず、システムSのサーバSaは、改版管理処理の移行をシステムTのサーバTaに依頼する。サーバTaが、文書の管理情報およびシステムTに登録されているサーバの一覧(Ta、Tb、…、Ti、…、Tm)を、サーバSaに送信する。サーバTaは、以降、文書の改版要求をサーバSaに転送する。この時点で、サーバTiが、文書4を1版から2版に改版したい場合には、サーバTiは、サーバTaに改版要求を発行するが、この要求はサーバTaによりサーバSaに転送され、サーバSaが要求を処理する。

【0118】サーバSaは、管理情報とサーバ一覧を受け取ったら、サーバSa、Sb、…、Si、…、SnとサーバTa、Tb、…、Ti、…、Tmとを統合した新たなサーバ一覧を作成し、この新サーバ一覧と改版管理

サーバのサーバS_aへの移行通知を、サーバS_a, S_b, ..., S_i, ..., S_nとサーバT_a, T_b, ..., T_i, ..., T_mに送信する。

【0119】各サーバは、サーバ一覧と移行通知を受け取ったら、自サーバのサーバ一覧を更新し、サーバT_aに対して確認通知を発行し、以降、その資源に関してはサーバS_aに要求を発行するようにする。この時点以降にサーバT_iが文書を更新しても、改版要求は、サーバS_aに直接送られるようになる。サーバT_aは、サーバS_a, S_b, ..., S_i, ..., S_nの全サーバから確認通知を受け取ったら、処理要求のサーバS_aへの転送を終了する。

【0120】

【発明の効果】本発明のドキュメント分散処理方法は、以下の効果を奏する。

1) 同一ドキュメントの複製を各サーバに配布しておくことにより、各サーバが任意の時点でドキュメントの更新を行うことができる。

【0121】2) あるドキュメントの更新を開始するサーバが他のサーバに対して通知を行い、受け取った他のサーバは更新終了までそのドキュメントに対する更新をロックすることにより、同一ドキュメントに対する多重の更新を回避することができる。

【0122】3) ドキュメントの更新開始と同時に他のサーバに対して、更新開始通知もしくは改版の承認申請を出し、承認を待つ間に更新を進めてしまうことにより、承認待ちのための手順および時間を省略することができる。

【0123】4) ドキュメントの実体であるファイルと、実体にリンクする論理オブジェクトとを分離し、データ量の少ない論理オブジェクトは新規作成・更新ごとに他のドキュメントアプリケーションサーバに反映させる一方、実体は、例えば定期的な時間間隔で複製を行うことにより、更新ごとに発生するデータ通信量を軽減することができる。

【0124】5) 上記4)の方式で、最新の実体を持つサーバの情報を保持することにより、各サーバが更新などにより最新の実体を必要とする場合に、それを保持するサーバから取得することができる。

【0125】6) 各サーバに優先度を設定することにより、別々のサーバで同時に更新を開始した場合に競合を回避することができる。

7) 各サーバごとに優先度に対応する独自のドキュメント版体系を設定することにより、仮に複数のサーバで同時に改版が発生してしまった場合でも、最も優先度の高い版体系を正式版として採用することで版の分裂を解決することができる。

【0126】8) 各サーバごとの優先度付き版体系を利用し、いずれかのサーバとの通信が途絶している場合でも残りのサーバ同士でドキュメントの更新を継続し、全

サーバ復活後に正式版に統合することができる。

【0127】9) サーバの優先度をドキュメント単位に設定することにより、1箇所のサーバへの通信集中を回避することができる。また、本発明のドキュメント分散処理システムのサーバ管理方法によれば、ネットワーク上の複数のサーバ間において同一のオンラインドキュメントを共有し、かつ、どのサーバにおいても参照・更新を可能とするようなマルチサーバによるドキュメント分散処理システム的环境において、同期によって他の処理が待たされることなく、かつ、一貫性を保持しながらサーバの追加、変更を行うことができる。

【図面の簡単な説明】

【図1】本発明の原理説明図である。

【図2】本発明によるサーバ管理のための構成を説明する図である。

【図3】文書の複製と配布を説明する図である。

【図4】送信処理および受信処理の処理フローチャートである。

【図5】第1の処理方式の手順の例を示す図である。

【図6】第1の処理方式における処理フローチャートである。

【図7】第2の処理方式の手順の例を示す図である。

【図8】第2の処理方式における処理フローチャートである。

【図9】第3の処理方式における文書の実装形態の例を示す図である。

【図10】第3の処理方式を説明する図である。

【図11】最新の実体の取得処理の処理フローチャートである。

【図12】第4の処理方式を説明する図である。

【図13】第4の処理方式における処理フローチャートである。

【図14】多重更新が発生する場合の例を示す図である。

【図15】第5の処理方式の手順の例を示す図である。

【図16】第5の処理方式における処理フローチャートである。

【図17】通信途絶が発生した場合の手順の例を示す図である。

【図18】第6の処理方式によるタイムアウトが発生した場合の処理フローチャートである。

【図19】サーバ一覧テーブルの構成例を示す図である。

【図20】第1の管理方式の手順の例を示す図である。

【図21】第1の管理方式における処理フローチャートである。

【図22】第2および第3の管理方式の手順の例を示す図である。

【図23】第2の管理方式における処理フローチャートである。

【図24】第3の管理方式における処理フローチャートである。

【図25】第4の管理方式の手順の例を示す図である。

【図26】第4の管理方式における処理フローチャートである。

【図27】文書を互いに共有するような業務を行うシステムにサーバxを追加する場合の実施例を示す図である。

【図28】それぞれ文書を共有するような業務を行うシステムSとシステムTとを統合する場合の実施例を示す図である。。

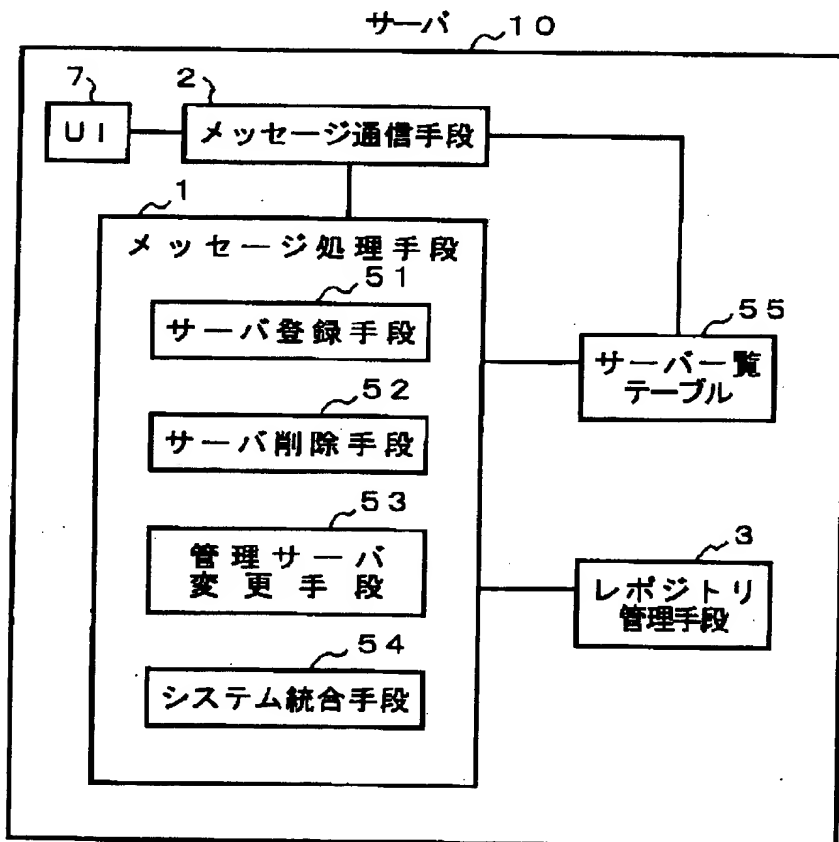
【符号の説明】

1 a, 1 b, 1 メッセージ処理手段
2 a, 2 b, 2 メッセージ通信手段

3 a, 3 b, 3 レポジトリ管理手段
4 a, 4 b 版数管理手段
5 a, 5 b ロック管理手段
6 a, 6 b 時刻監視手段
7 a, 7 b, 7 ユーザインタフェース (UI)
10 a, 10 b, 10 サーバ
20 a, 20 b レポジトリ
30 a, 30 b クライアント
40 ネットワーク
51 サーバ登録手段
52 サーバ削除手段
53 管理サーバ変更手段
54 システム統合手段
55 サーバ一覧テーブル

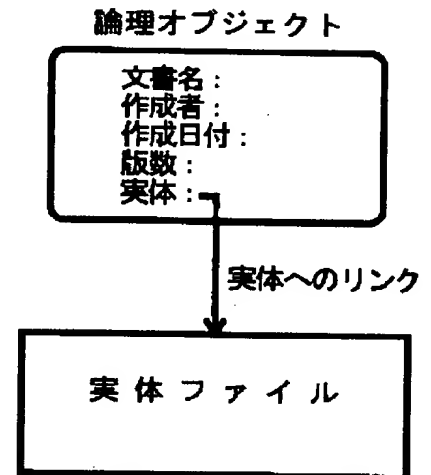
【図2】

サーバ管理の構成説明図



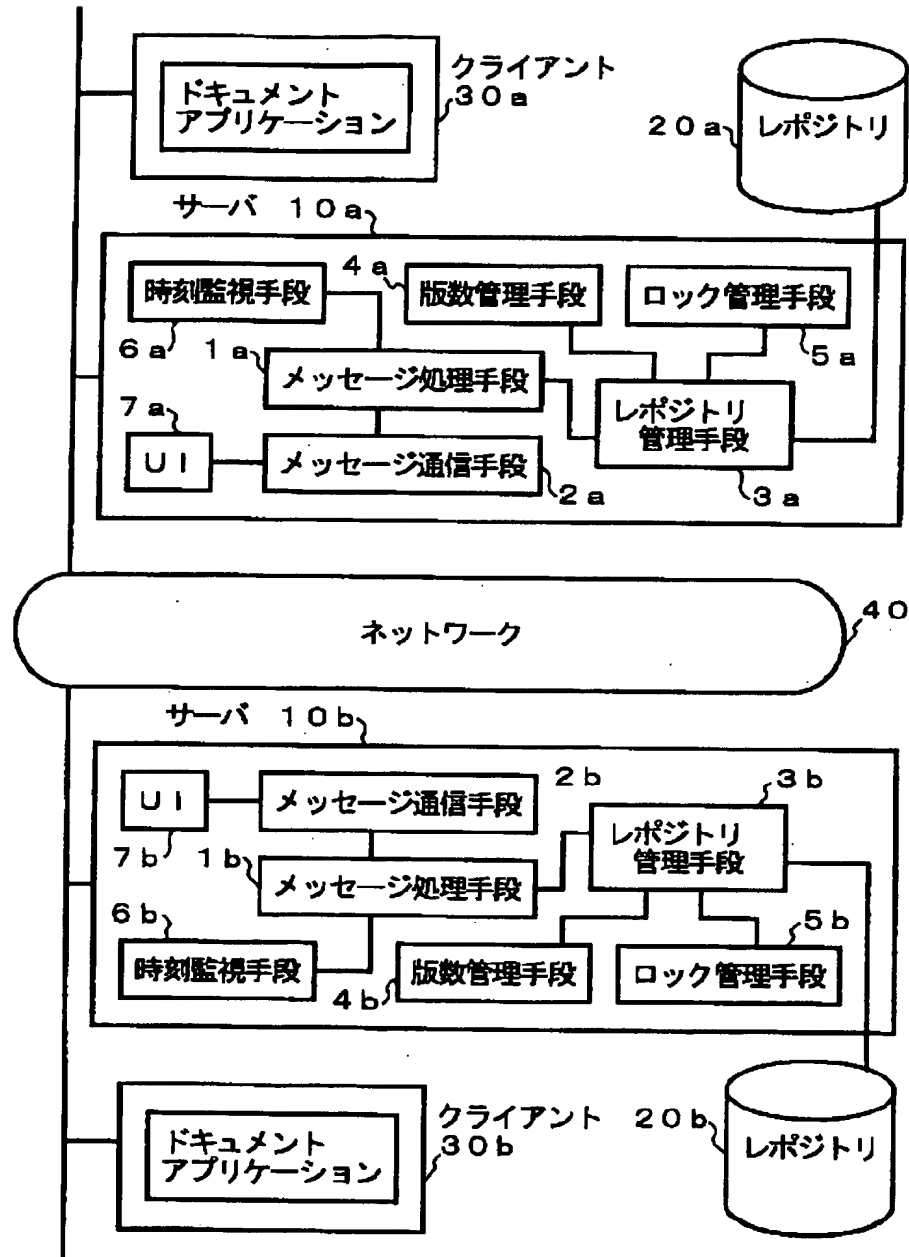
【図9】

文書の実装形態の例



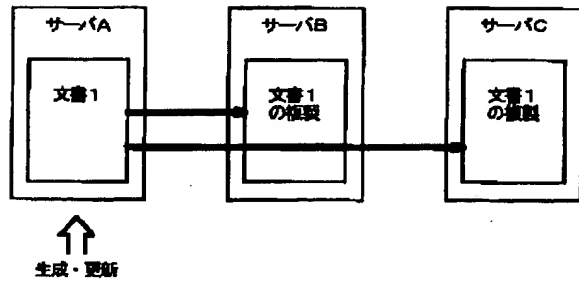
【図1】

本発明の原理説明図



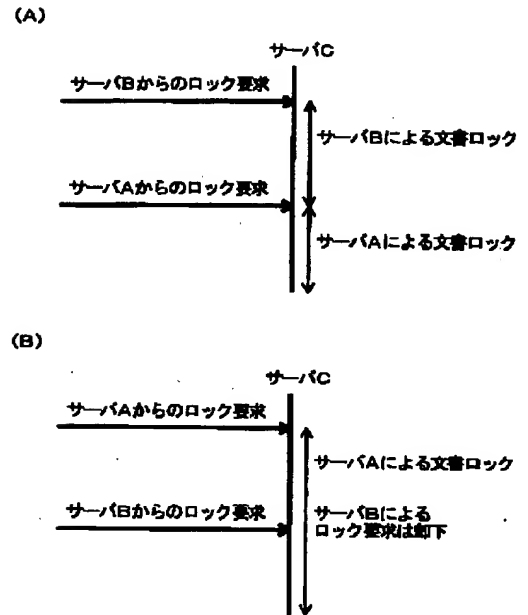
【図3】

文書の複製と配布



【図12】

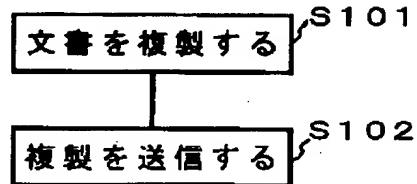
第4の処理方式の説明図



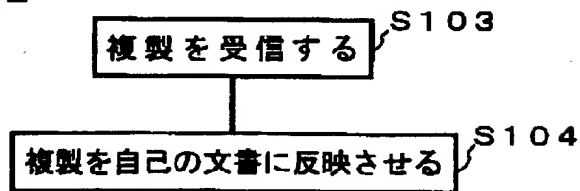
【図4】

送信／受信処理の処理フローチャート

(A) 送信処理



(B) 受信処理



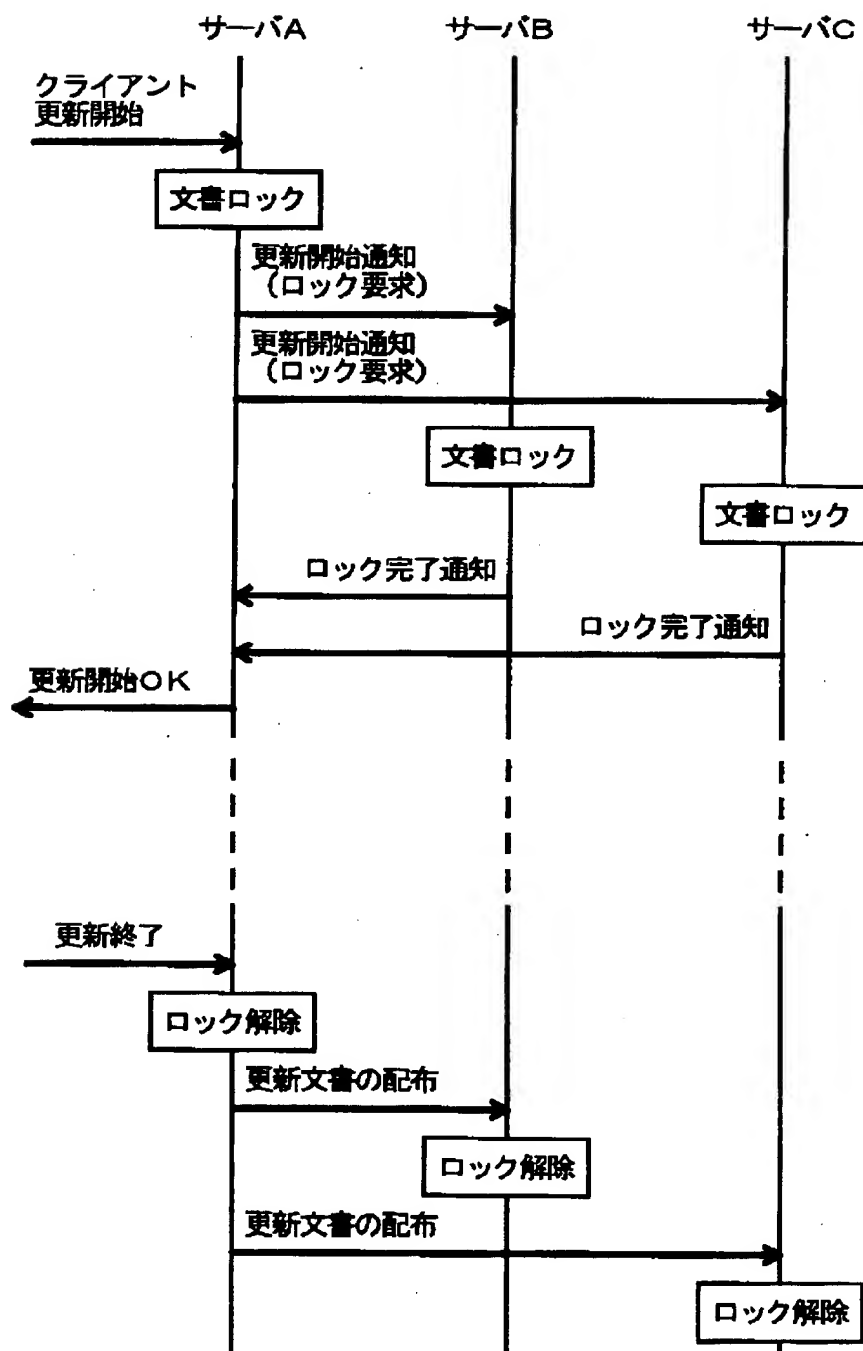
【図19】

サーバー一覧テーブルの構成例

サーバ名	宛先 (アドレス)	属性
⋮	⋮	⋮

【図5】

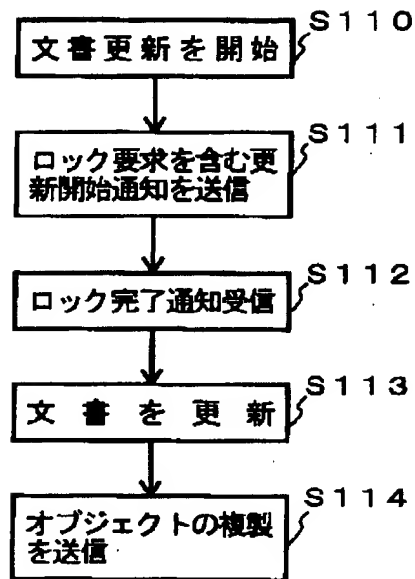
第1の処理方式の手順の例



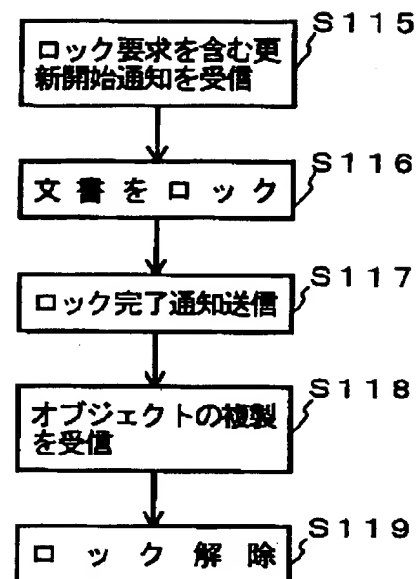
【図6】

第1の処理方式における処理フローチャート

(A) 送信処理

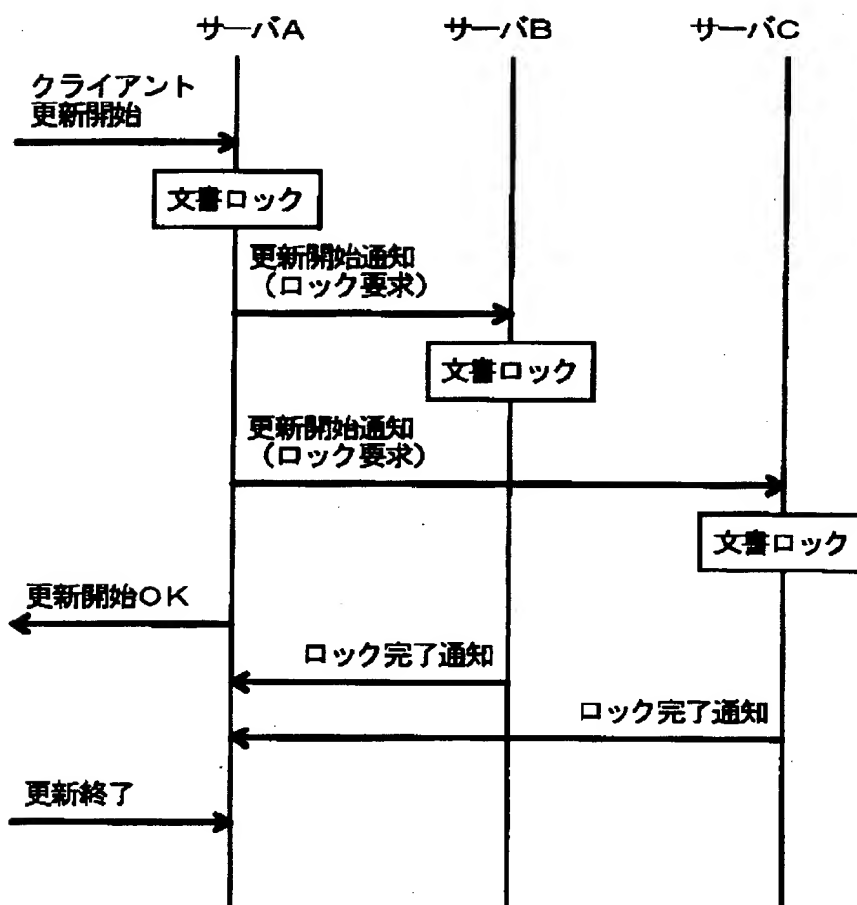


(B) 受信処理



【図7】

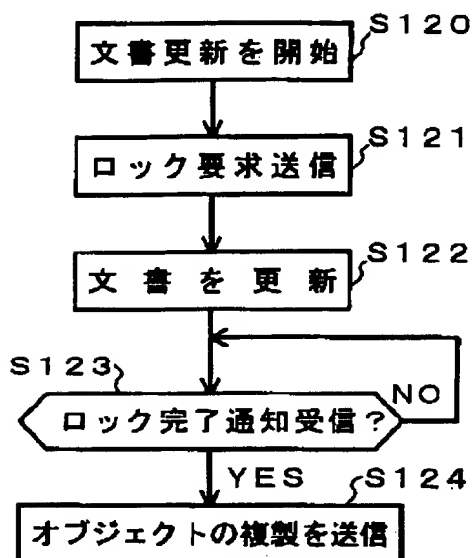
第2の処理方式の手順の例



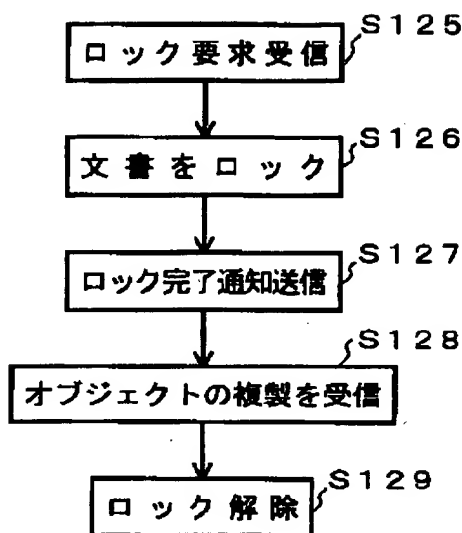
【図8】

第2の処理方式における処理フローチャート

(A) 送信処理



(B) 受信処理



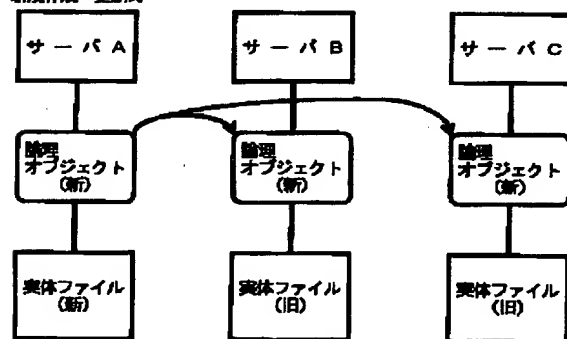
【図10】

【図20】

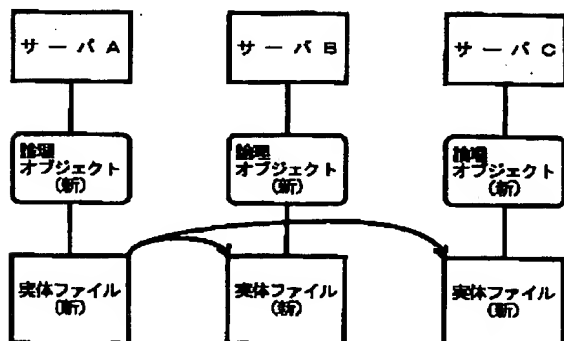
第3の処理方式の説明図

(A)

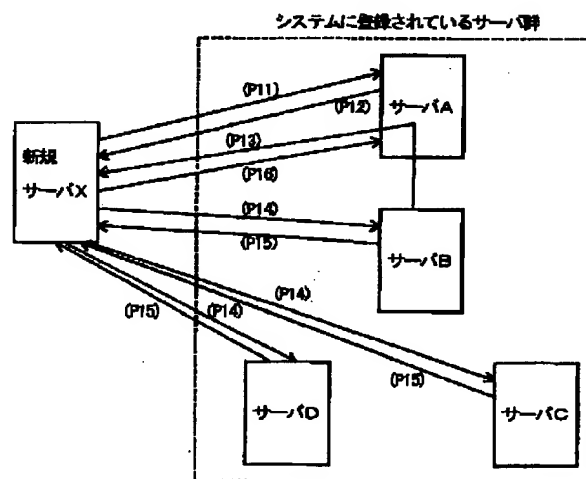
新規作成・更新時



(B)



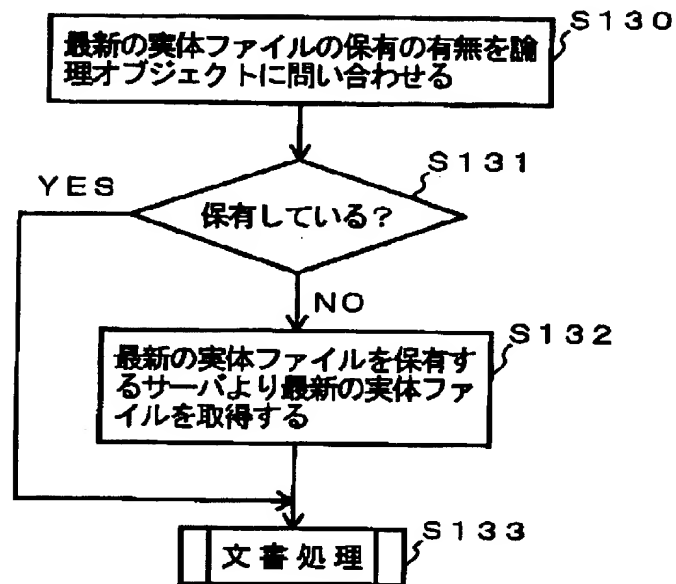
第1の管理方式の手順の例



- (P11) : サーバ一覧 データの申請
- (P12) : サーバ一覧 データ送信
- (P13) : 更新されたデータの転送
- (P14) : サーバの登録の依頼
- (P15) : 登録完了通知 データ転送
- (P16) : 転送終了依頼

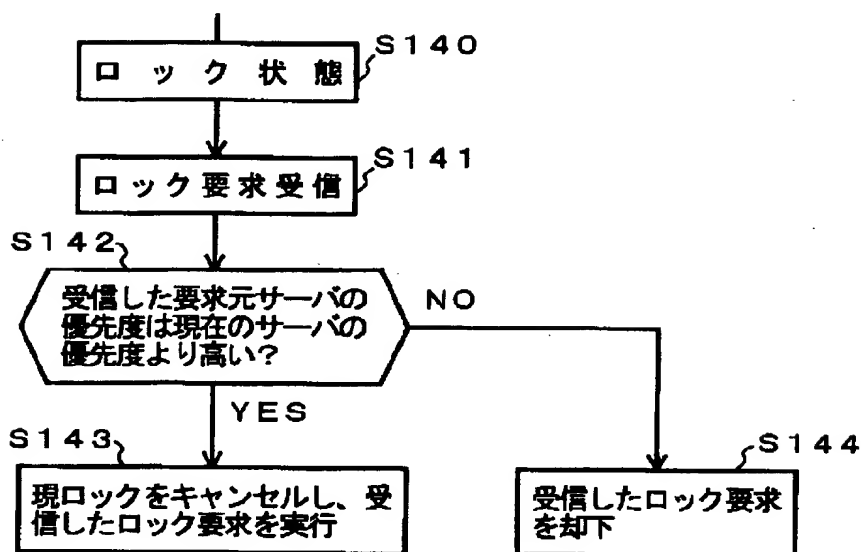
【図11】

最新の実体の取得処理の処理フローチャート



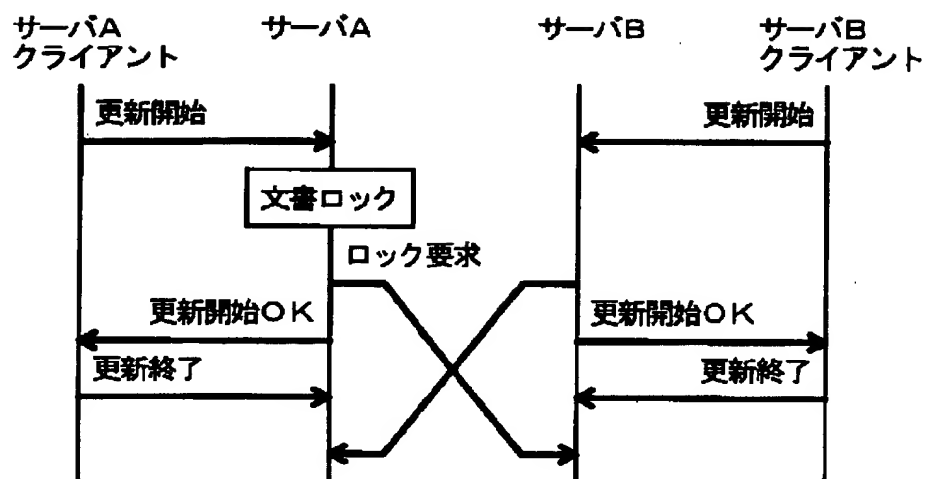
【図13】

第4の処理方式における処理フローチャート



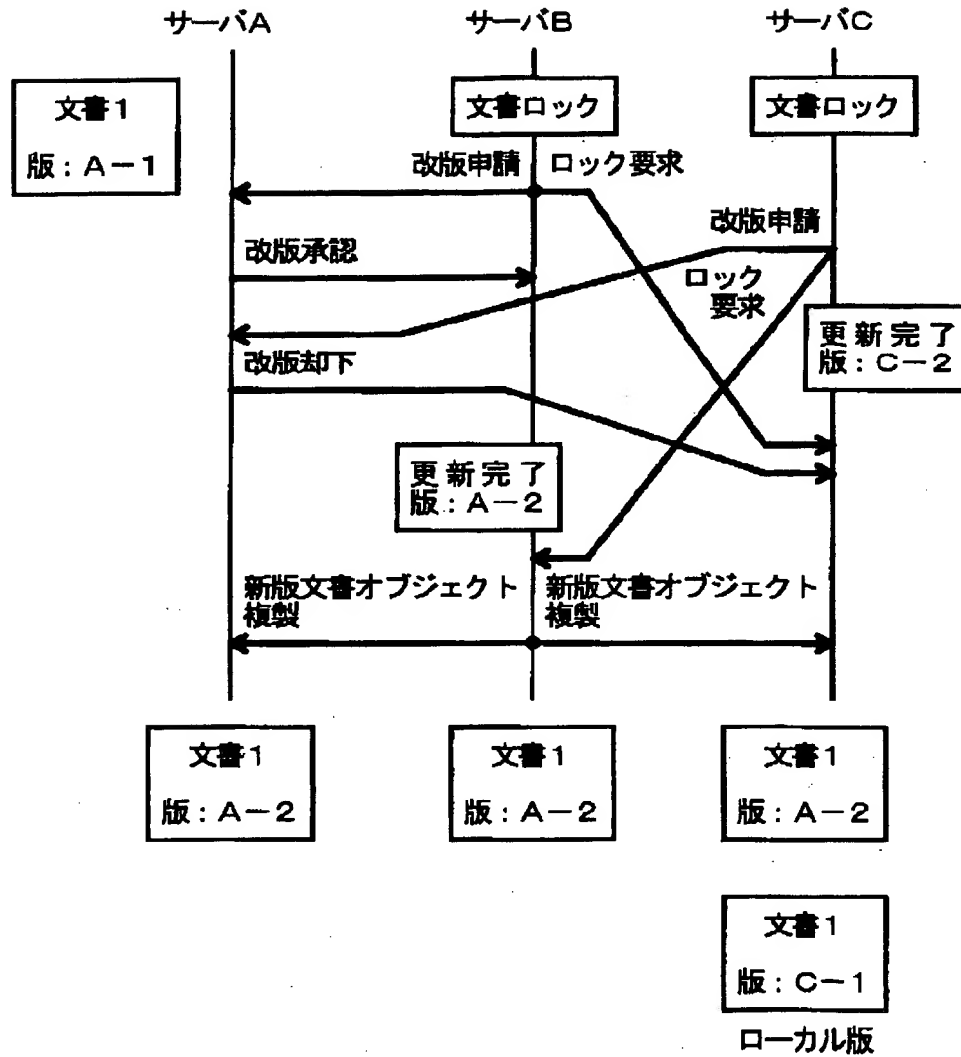
【図14】

多重更新が発生する場合の例



【図15】

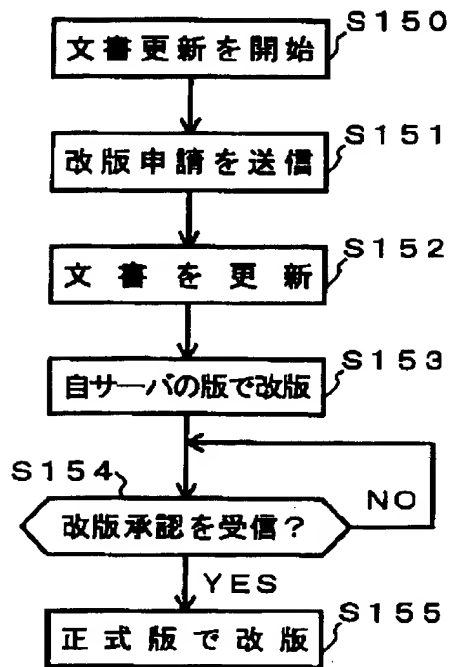
第5の処理方式の手順の例



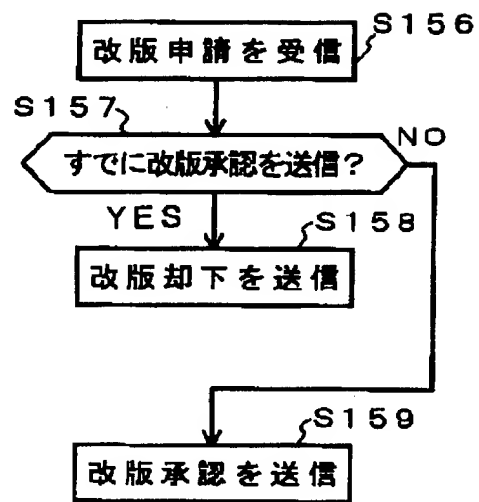
【図16】

第5の処理方式における処理フローチャート

(A) 更新サーバ

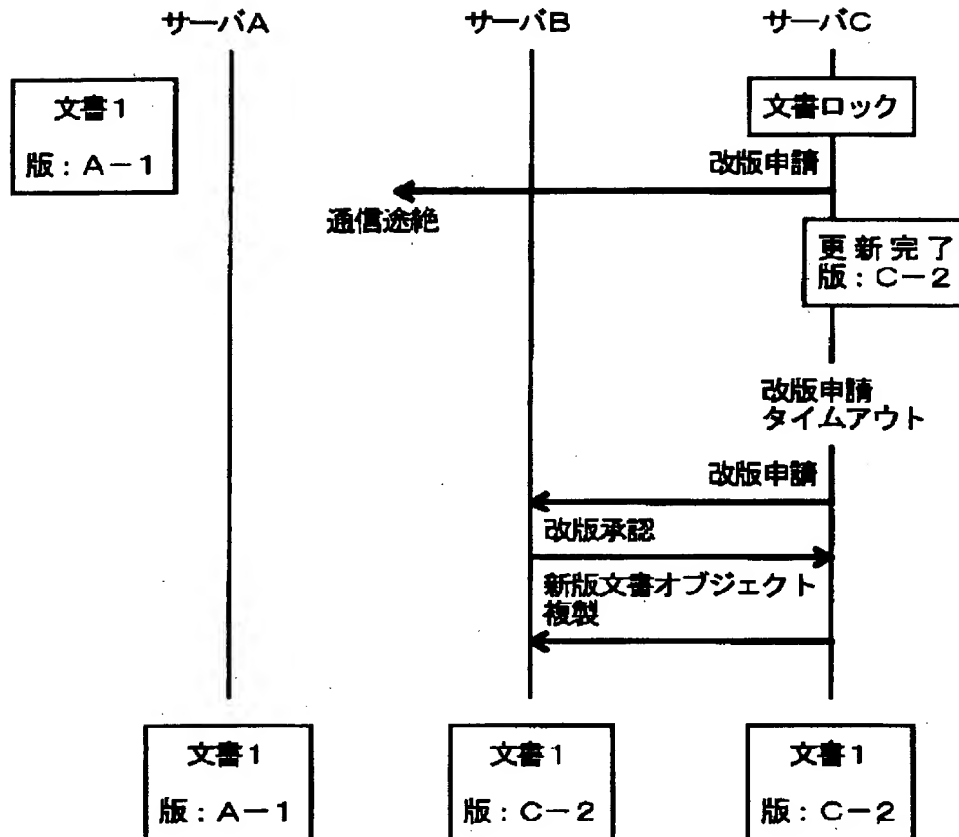


(B) 正式版発行サーバ



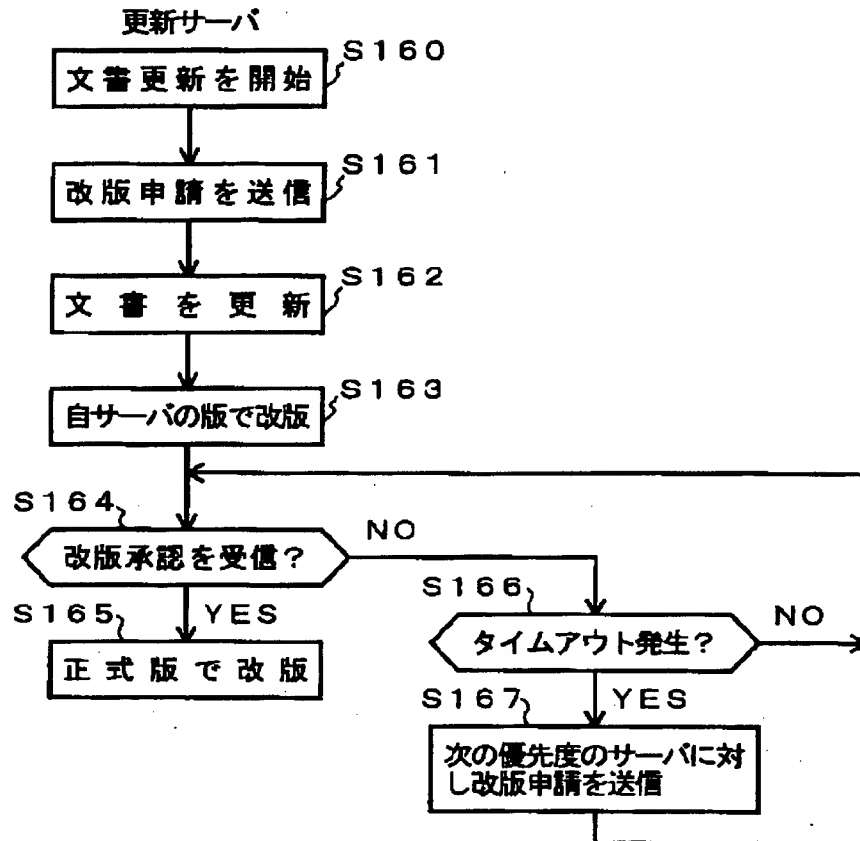
【図17】

通信途絶が発生した場合の手順の例



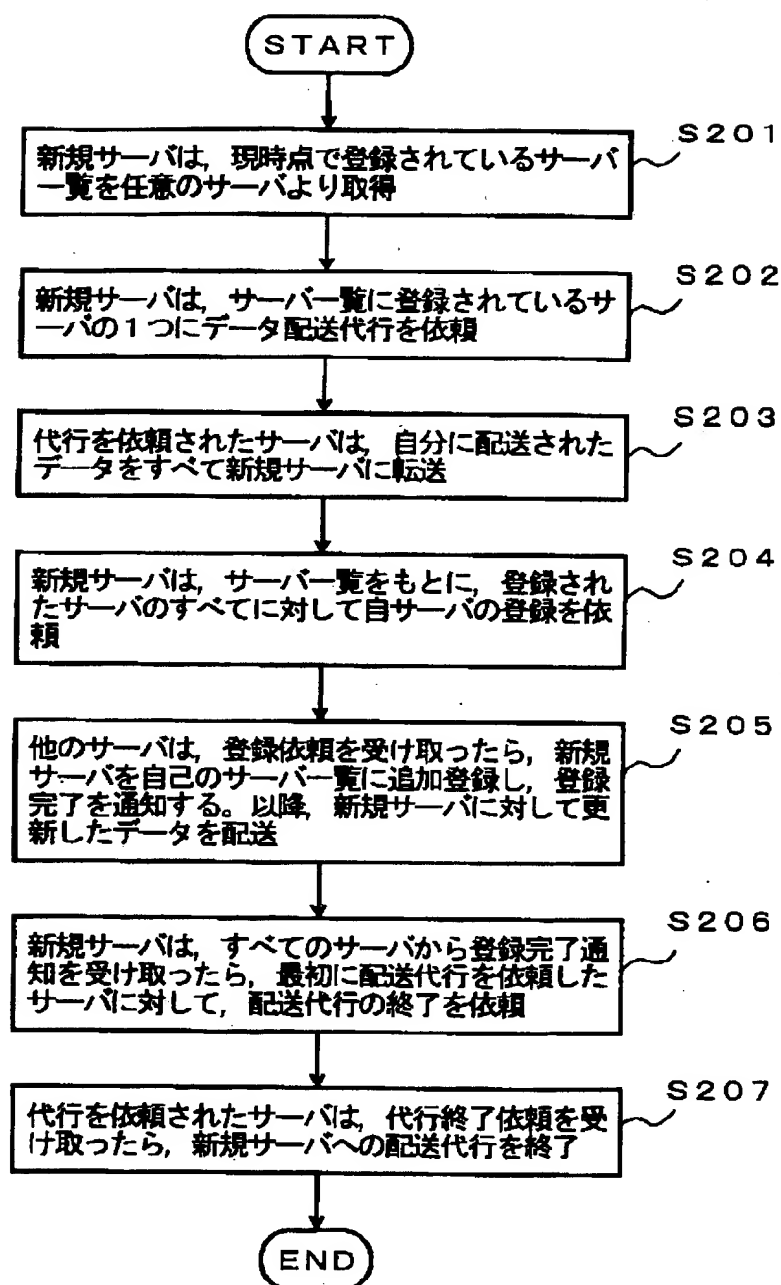
【図18】

タイムアウトが発生した場合の処理フローチャート



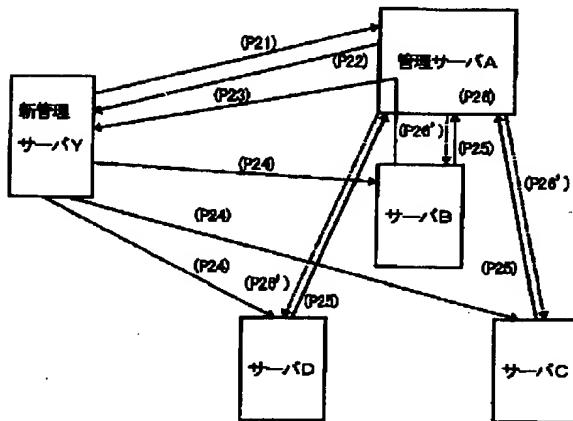
【図21】

第1の管理方式における処理フローチャート



【図 2 2】

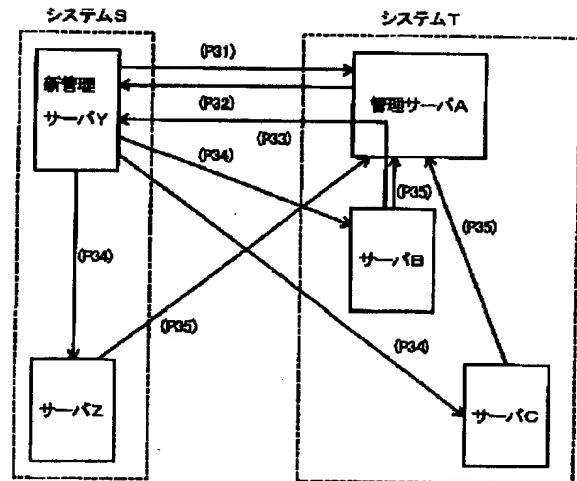
第 2 および第 3 の管理方式の手順の例



- (P21) : 移行申請
 (P22) : 移行の承認と管理情報の送達
 (P23) : 要求の転送
 (P24) : 移行通知
 (P25) : 確認通知
 (P26) : 転送終了 (第2の管理方式)
 (P26') : 転送終了依頼と管理情報の更新 (第3の管理方式)

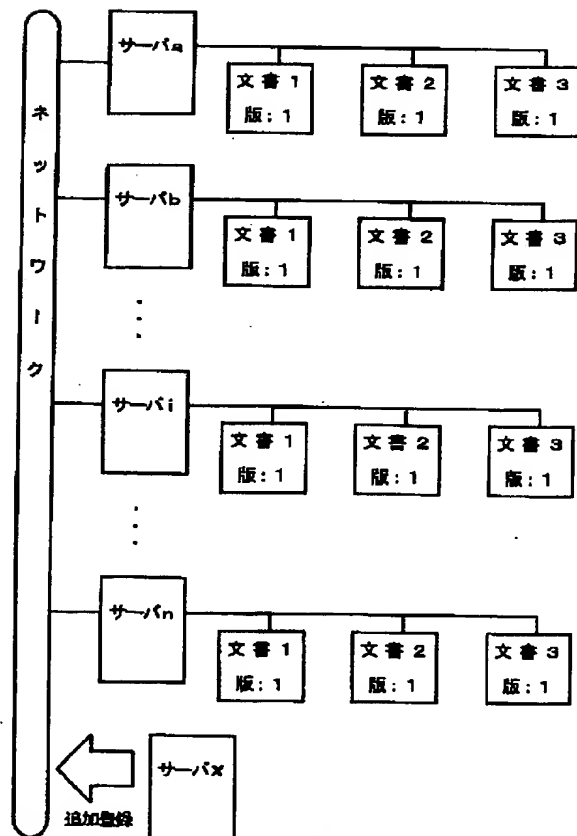
【図 2 5】

第 4 の管理方式の手順の例



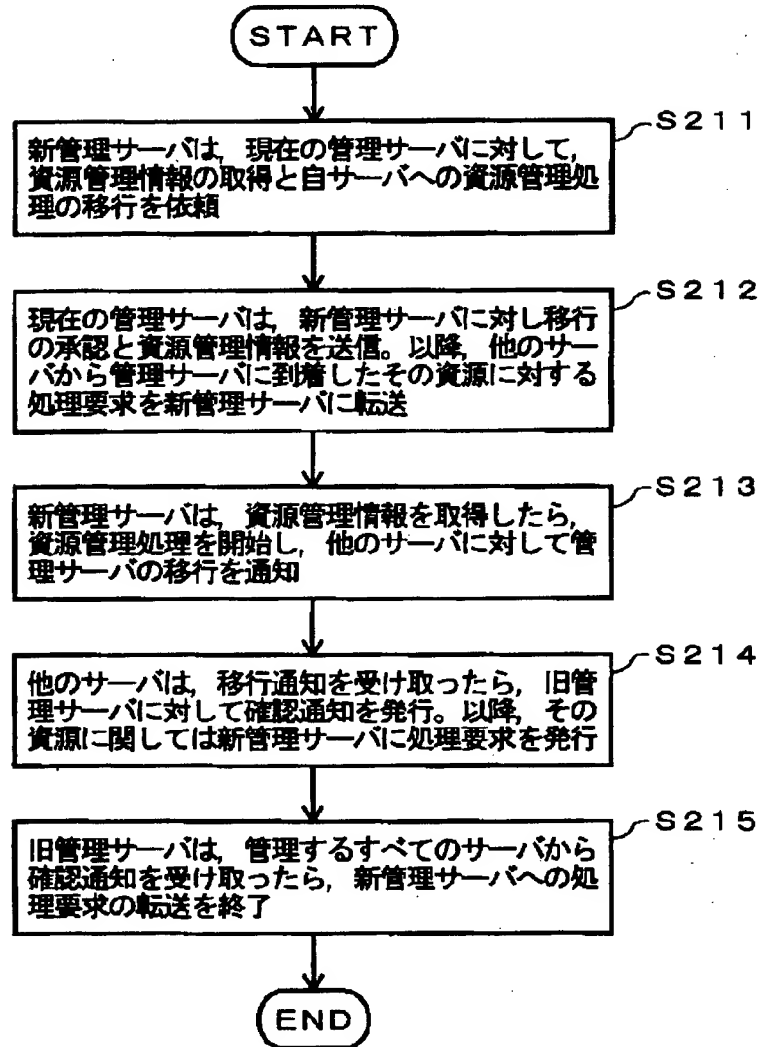
- (P31) : 移行申請
 (P32) : 移行の承認と管理情報、サーバ一覧の送達
 (P33) : 要求の転送
 (P34) : 移行通知と新サーバ一覧の配送
 (P35) : 確認通知

【図 2 7】



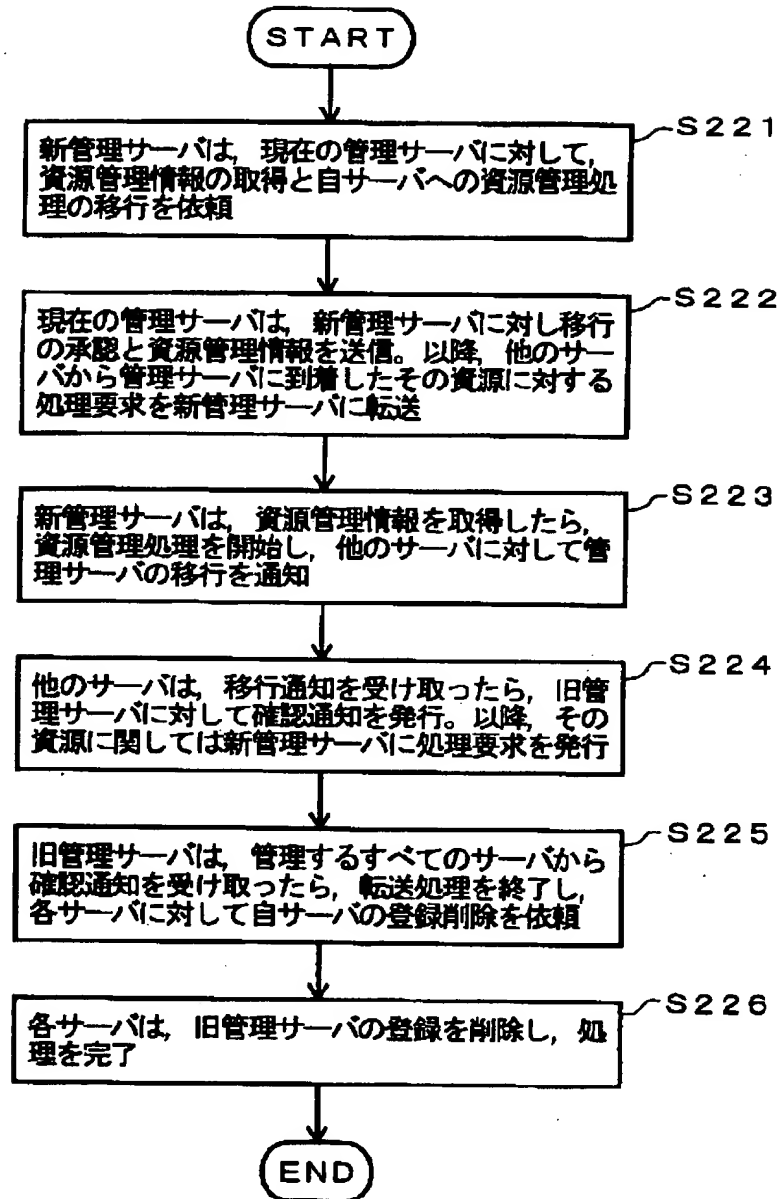
【図23】

第2の管理方式における処理フローチャート



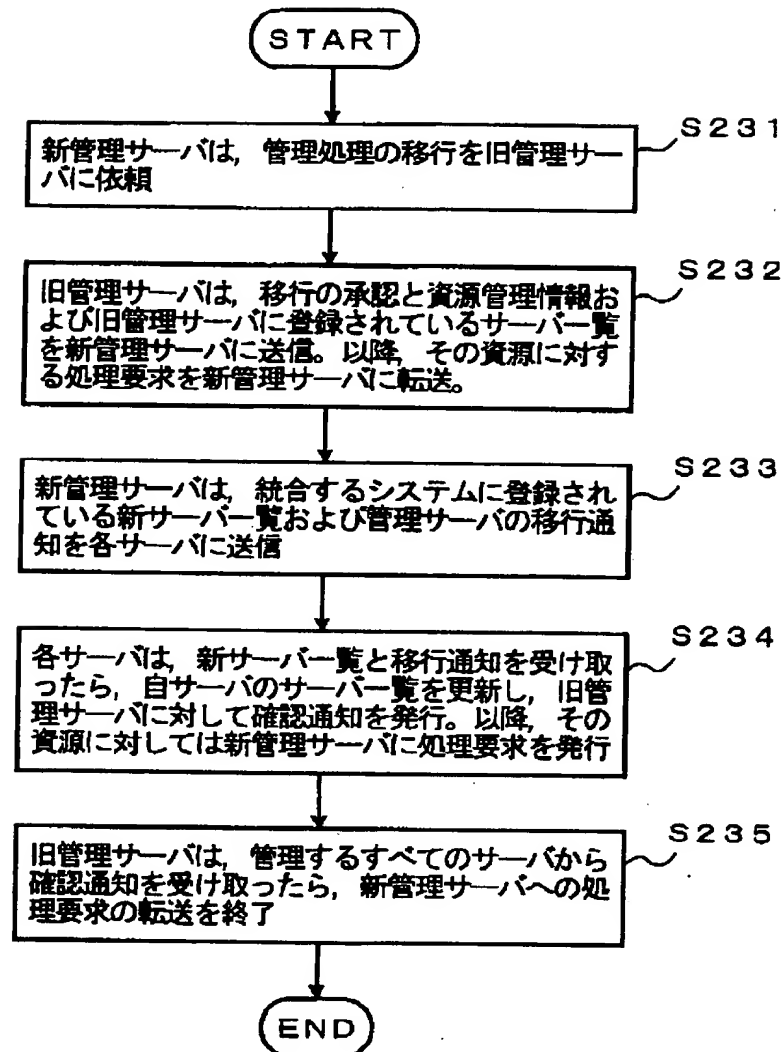
【図24】

第3の管理方式における処理フローチャート



【図26】

第4の管理方式における処理フローチャート



【図28】

